THE CAUSAL HYPOTHESIS

REALIZING THE POTENTIAL OF PROCESS MINING



noreja process intelligence

Table of Contents

Introduction 3

The Trouble with Transformers 5

- Cost of Generation 6
- Time per Iteration 8
- 2D Conversion 9
- Why the Trouble, Then? 10

The Way Forward 11

- Is OCPM Enough? 12
- Why hasn't this been done before? 14
- Introducing Causal Process Mining 15
- Novel, but Grounded in Precedent 18

Applying the Theory 20

- Optimizing the Process Mining Process 20
- True Process Flows, Without the Spaghetti 22
- Finding Out if the Beaten Path Needs Guard Rails 23
- Complex Evaluations and Predictive Modeling 24

Growing Pains 25

- Environmental Challenges 25
- Performance and Technical Challenges 26

Table of Contents

The Steep Slope Up 28

- Neural Network Compatibility 28
- Automation Embedding 29
- The One-Stop Shop Data Lake 29

Conclusion 30

Authors 32

Sources 33

Introduction

When the world was introduced to process mining in 1999, the internet was still in its infancy, mobile phone adoption rates were still in the single digits, and cable television was going from strength to strength, unaware of the digital revolution looming just over the horizon. In essence, the world was just beginning to discover what digitalization even means, with the world's first online payment provider only having been founded a year prior[1]. Even then, the value of the technology was evident. Being able to trace digital events in businesses and enrich them with their process flow relationship information is one of the first steps towards the true predictive engines of science fiction fever dreams, after all.

Now, some 25 years later, we live in a very different world. Most people in the developed world possess at least two fully internet-integrated devices, spending a worldwide average of six and a half hours interacting with a digital screen every day[2]. All kinds of data are abundant across every industry segment and demographic and it's widely accepted that leveraging this data is a key success factor in most, if not all industries. Yet, the tools most organizations have available feel limited and incomplete. Business intelligence solutions are ubiquitous, but only show part of the picture, and our prospective first step into that fantastical future, process mining, despite growing healthily as a market every year, still feels underrepresented and underappreciated.

We believe that one of the largest contributors to this situation is the implementation approach, in large part, remaining unchanged over these last two and a half decades. Just like the **Alpha Miner** research prototype in 2001[3], almost all providers in the space still rely on event logs and flat data structures to generate their process models, despite the significant costs and disadvantages[4] that come with them.

We're not the only ones that think this way, as can be evidenced by some of the more innovative players in the market moving on to more modern, novel techniques, all surrounding the concept of object-centric process mining. However, these innovations have begun happening only in the last two to three years, and in our opinion, they aren't sufficient to catch the field up to the modern, digitalized era and unlock the true, full potential that it can offer the world of data science and process analysis.

It is for this reason that we are putting out this paper – to flag what we see as the limitations and restrictions that the process mining community currently takes as a given, and to propose that these are not limitations that we have to live with, but rather growing pains that can be shed and left in our rear-view mirror as we move towards a future of limitless potential.

First, we will deconstruct what the current problems the industry is facing are, why they were necessary for it to grow to its current maturity level, and why they should no longer be something we take as given. Then, we detail our new approach to process mining, which we call **Causal Process Mining**, and explain how it fulfils the requirements of object centric process mining better than current market solutions, as well as the novel use-cases that are unlocked via this approach. Finally, we'll look at the future and discuss both the challenges and difficulties facing the implementation of this approach, as well as the massive future potential we see when these difficulties are overcome, and our approach reaches the level of maturity that the current market standard enjoys.

So - go grab a coffee or some popcorn and join us on this voyage through what was, is and will be in the world of process discovery and analysis. **We promise it's going to be an exciting ride.**



The Trouble with Transformers

Why Event Logs slow us down

Current, market-standard process mining approaches have, up until now, almost exclusively utilized **event logs**. For those not already in-depth on the details involved in process mining, what this means is that to get started with a process mining solution, companies have currently first needed to convert the raw data in the source databases into a flat-file structure[5], containing the following attributes:

Case Identifier

The ID that acts as the global identifier throughout the data, e.g. the Order Number for a Commerce-centric process.

Event

The description of what happened to the case at every step, e.g. Order Placed or Shipment Sent.

Timestamp

The time and date of the event in question.

Context Data

Any supplementary or supporting information from the source dataset, e.g. Customer Location or Order Volume.

This approach takes a tremendous amount of time and effort just to get the tool operating, and all of this is before we even truly know if the tool can help our organization. And even after going through all of this, it generates a skewed, filtered view of our true business reality, impairing our ability to view real connections between events, whether causal or otherwise.

Cost of Generation

While it's hard to estimate the exact costs for generating an event log across the board, we can make some basic assumptions here to help us understand the problem. First, let's split this calculation into two steps: the initial implementation and the maintenance and modification of the log in response to changing business needs. After all, any good software solution needs to adapt to changing real-world circumstances to always remain current. One survey found that **93% of its sample** agreed or strongly agreed that this step of the process was the most significant portion of effort involved in implementing a process mining solution at an organization[6].

Let's take a look at why this is. First, consider the initial setup of the event log. Generally speaking, you need at least **four people** for this process to work well:



Process Mining Consultant

Usually External

Uses Process Mining Knowledge to steer the process according to technical needs



Business Owner

Dept. Head or Executive

Determines which strategic goals should be evaluated for and which processes should be looked at.



Process Manager

BPMN Analyst

Understands the existing processes in the organization and owns the intervention process.



Systems Administrator

Data Owner

Has accesses to the data repositories of the organization and can provide the requisite data.

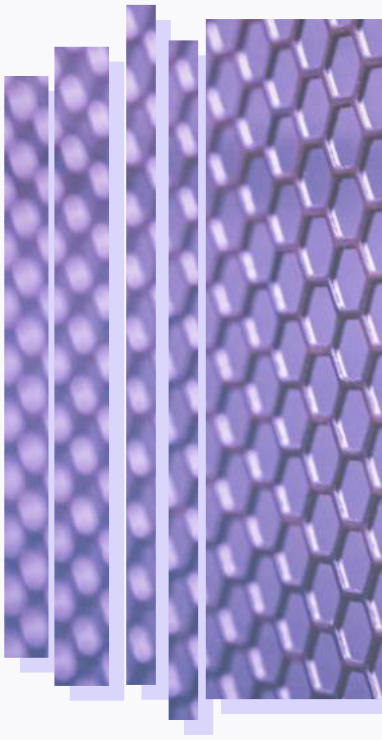
The process usually involves aligning the kind of data that needs to be included in the event log with business needs and mapping the existing understanding of processes to specific data traces in the organization's systems. This is then used to either directly create an initial, manually created event log or codified as a transformer script that can continuously generate event logs from the defined dataset.

Because of the complexity involved in this, the varying states, locations and quality of data, the number of alignment and decision-making cycles and the manual work in building this transformer, this process can take as long as 2-4 months, with some estimates of real cost being **just under €100,000**[7] in both consultant and internal people-hours – just for the initial setup of the tool.

This effort needs to be repeated – with no guarantee of economies of scale and repetition – for every new process that we want to implement process mining for. Large organizations whose businesses benefit enough from process mining that they can build up their internal departments to handle this need may be able to absorb these costs and justify them, but for smaller to medium sized organizations, this all but **precludes the use of process mining at scale**.

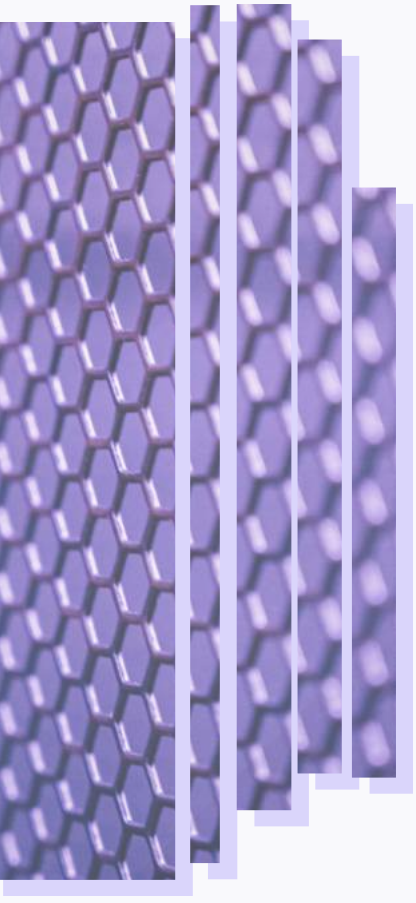


Time per Iteration



If things are missing in our initially generated event log, if there are mistakes in the transformer, or for whatever reason changes are needed, this will require **going back to the previous process** and getting all the same parties involved. As the transformer needs to be rebuilt or a new event log needs to be manually generated, every one of these modifications incur both monetary and time costs.

Of course – we expect changes to be needed over the lifetime of a tool, not only to reflect our business environment changing, but also as we change our processes in response to the findings in the tool. In this regard, we can say that fluidity and modification are expected and desired properties in the mid- to long-term use process mining.



It stands at a perpendicular, then, that the prerequisite for using process mining applies high costs to exactly this iterative approach – the more we use the tool for business agility and transformation, the more we need to change its core parameters, **the more costs it accrues for every experiment**. This creates a financial barrier to following the spirit of innovation and iteration, instead offering a pull to “stick with what you have”.

Worse, it’s a widely accepted concept that the later in the lifecycle of a project you discover the need for a change in its approach, the more expensive that change is and the harder it is to perform[8]. This applies to change management as well, and with the need to change your event log transformers to adapt to new circumstances, the time it takes to get feedback if your modified process is performing better or worse increases, thus accruing new, secondary costs, and **reducing the success rate** of your process management initiatives.

2D Conversion

Perhaps the most-discussed issue with traditional process mining has so far been its inability to distinguish between concurrent events – the root cause of traditional process mining models generating what is called “spaghetti diagrams,” as popularized by Wil van der Aalst, one of the founding fathers of Process Mining[9]. The core of the problem lies in the conversion of inherently **multi-dimensional organizational data realities** into an inherently **two-dimensional format** as part of the event log transformation.

Because we have a single-threaded path through a traditional event log, we lose the ability to distinguish between two similar events occurring concurrently, leading to false process transitions that don’t really exist in the business reality, artificial self-loops where a single event appears to repeat on itself, and other spurious connections[10]. Generally, this is the joint problems of **process-adherent** and **process-deviant behaviour** and of **convergence-divergence**.

One artefact of this conversion approach is a loss of data that occurs when preparing event logs. Since event logs need to be internally consistent and create complete models, a common practice is to remove outliers and complex process instances from the log prior to analysis[11]. Additionally, most traditional process mining tools offer and encourage the use of filtering out full outlier cases to reduce the noise and visual complexity created by spaghetti diagrams. Both approaches lead to purposeful, encouraged blind spots in your analysis as a method of making better sense of the results.

Another problem is that the resultant logs and flat files are highly use-case-specific. Just like that expensive egg-slicer that’s undoubtedly lying in many readers’ parents’ kitchen drawers, the proportionally even more expensive data structures created by this approach can only be used for one thing – process mining. On the surface, that seems fine – this is its purpose, after all. However, it leaves a lot of value on the table – ideally, if we’re spending time and money generating digital representations of the complete reality of a business, we would like to be able to use it for other analytical purposes as well. We’ll talk about this in more depth later.

Why the Trouble, Then?

So, if this approach is so flawed, why has it been the market standard? Well, flat data structures like event logs have some advantages that were key to launching process mining as a field in the 2000s:

- **Event logs are performance-efficient:** compute power was much more limited when the field was launched. Now, after 15 years of Moore's Law, we can handle much bigger and more complex datasets.
- **ETL as Part of the Core Process:** Data quality over a decade ago was significantly lower than it is today, so rigorous ETL processes and manual labor were intrinsically necessary to perform process mining; event logs as an output fit naturally into this process.
- **An Easy to Implement Standard:** With data science being a much less common field of practice at the time, having a simple standard for an output document was advantageous for early adoption of these tools in a nascent market.

These advantages were necessities for kickstarting the practice, and we think it's safe to say that without this approach, the market and field of study would not have taken off when it did. However, we also think, just as it was time in 2007 to shed the constraints of the feature phone and usher in the new era of the futuristic personal terminal in the form of a smartphone, it is now time to shed the constraints of the event log and move forward with a smarter, faster and more future-oriented approach to process mining.

The Way Forward

Causal Process Mining and Direct-to-Database

The Saviour du jour as the Process Mining market matures and sheds the fault-tolerance that has accompanied its nascence – the faults were a solid trade-off, after all, for the novel evaluations that it enabled – is **object-centric process mining**, or OCPM. Positioned as a revamp of traditional process mining, OCPM attempts to both bring the third dimension back into process models, fixing convergence-divergence issues and other artifacts, and provide the ability to connect processes from multiple distinct business areas into one network.

This is sorely needed, and wonderful addition to the landscape, but we believe that the current approaches to achieving this are not enough to fulfil the goals set forth by thought-leaders like Wil van der Aalst[1][2]. Our proposed alternative is, strictly speaking, also object-centric, but with one key addition. We shed the limitations of two-dimensional, flat data structures and embrace the **networked reality of business processes**, enabling a greater level of detail and novel evaluation approaches. In this section, we'll go through what our proposal entails and why we believe it to be an evolution of the field.



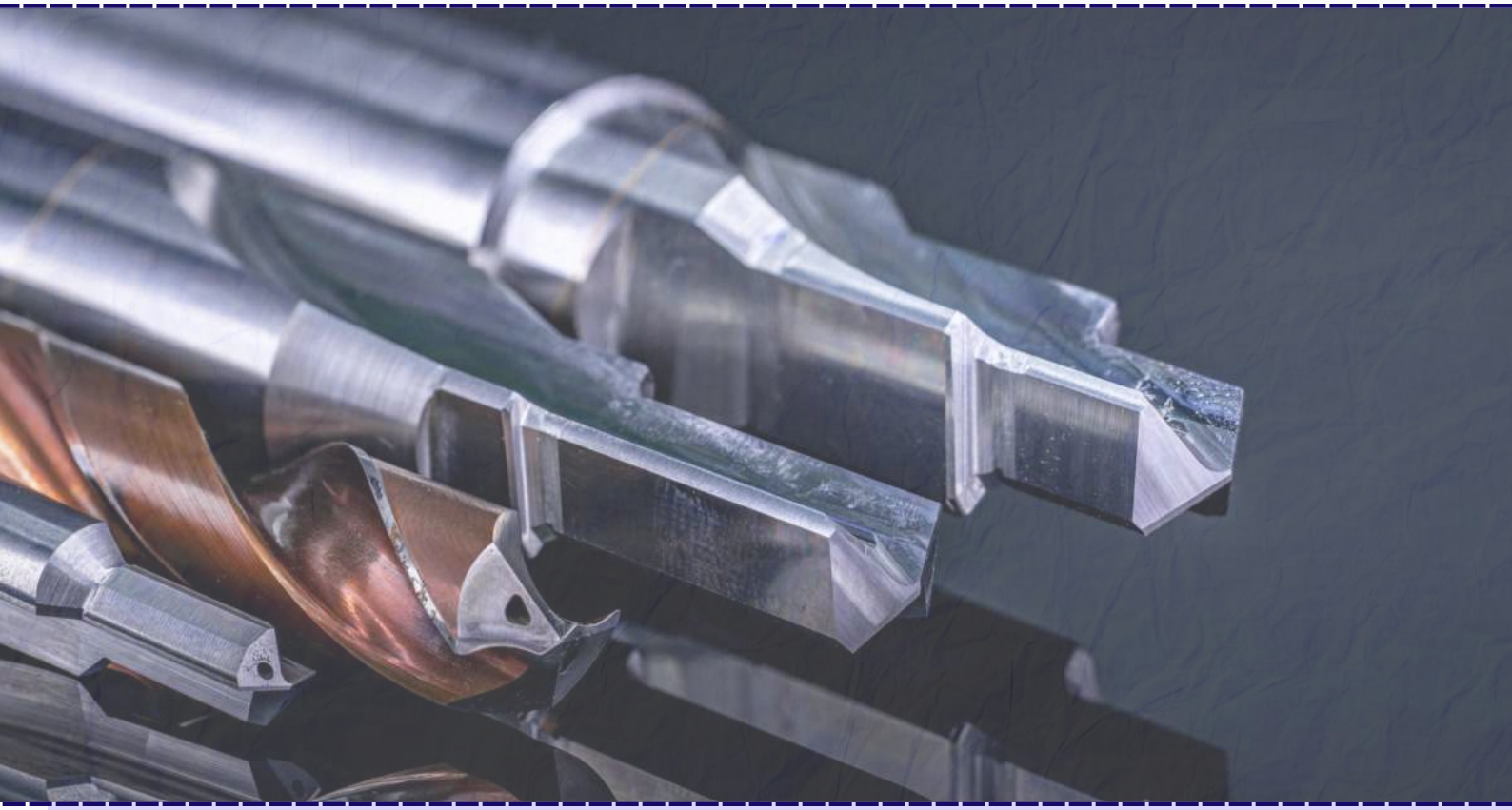
Is OCPM Enough?

Two major approaches exist for implementing OCPM in the market. The first, the **multi-log approach**[13], attempts to navigate and mitigate the "flattening" issue by using multiple event logs from the perspective of different objects in the organization. For example, a shipment may contain items from multiple orders, which with traditional single log process mining would create self-loops and cross-links that didn't happen in the business reality. By using a log that focuses on the shipment objects as a secondary reference and consistency check for the process map, this approach can identify and remove self-loop artifacts and simulate concurrent events.

There are three problems with this approach. First, it layers **imperfect inferences** as to the relationships of events without considering causality. While layering these inferences can improve the evaluation, it can't achieve true causal modelling and, and in the worst cases, it can create compound artefacts. Second, it requires prior knowledge about the clustering of objects and batched work in your organization, which at best requires extra manual discovery steps, and at worst creates blind spots instead of eliminating them. Third, every additional object-centric log that you need to add to the mix requires you to **create that log** – which, as we already covered, is a costly and time-intensive endeavour.

A more modern approach is to work with true object-centric event logs that build on relational database data structures[14]. This is already a step in a much better direction, as it at least attempts to mimic three-dimensional data, but there are still significant limitations to this approach. For one, in a very general sense, relational databases are simply not good at dealing with the kind of networked data that process events create[15]. For another, this approach has **no intrinsic handling of process compliance and adherence**, building on inferring what triggers what from the data. There are methods that add this functionality on top, but this is an additional step – we believe in a more elegant, integrated approach, as we consider this to be core to the process mining use case.

For both approaches, the egg slicer or “unitasker” problem of the traditional event log persists. Object Centric Event Logs and Multi-Log Mining both create data that is specialised for this single analysis use-case, requiring **new data extraction** even for another process mining use-case. They cannot be used as starting points for another analysis.

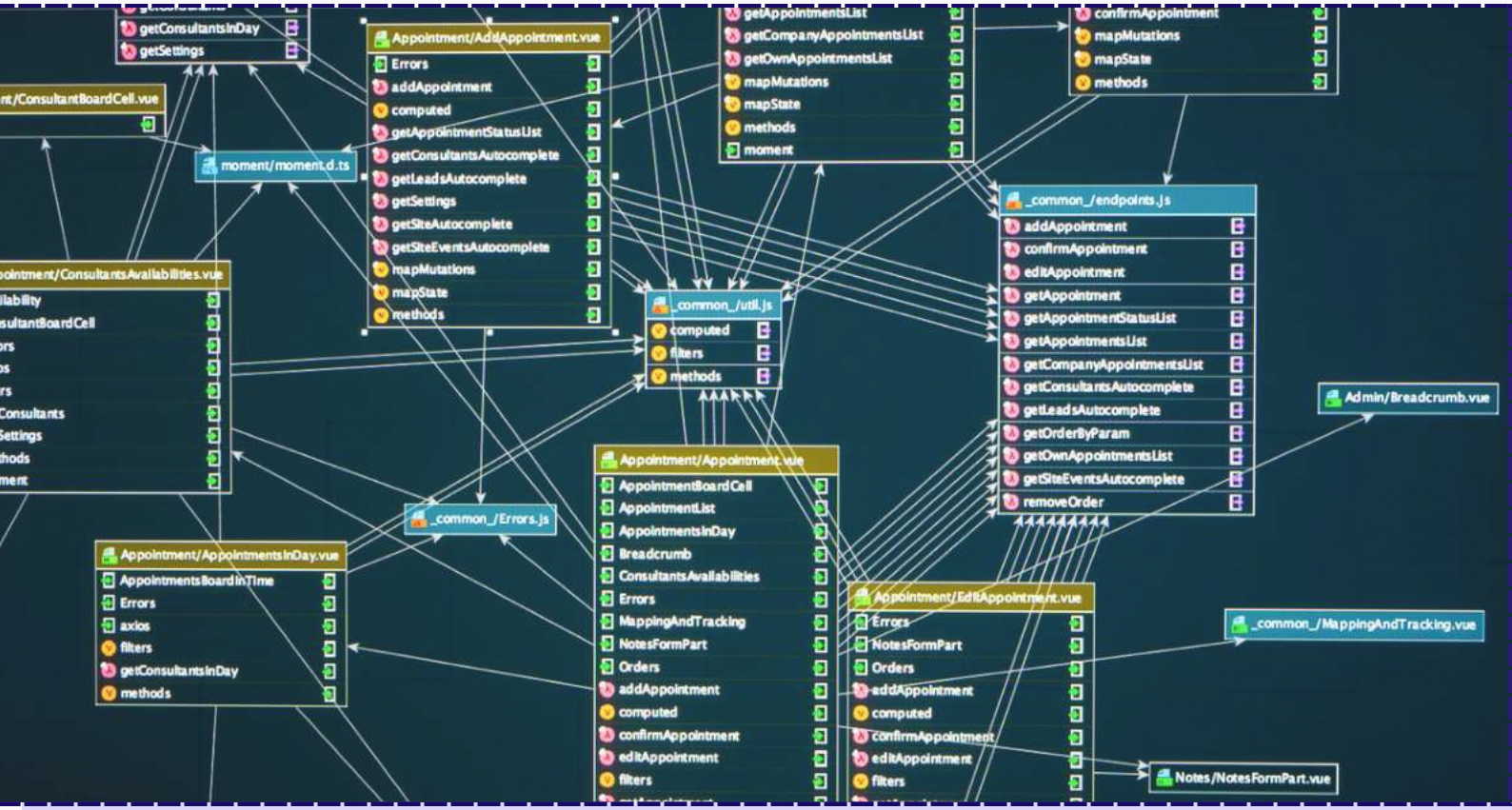


We asked ourselves – why? Why not take three-dimensional data and put it directly into a three-dimensional format, leaping over the step of creating an event log? The enterprise software world **“down samples” business reality**, reducing its definition to fit it into flat data structures, keeping only the relations necessary for the client ERP or CRM to function. But what interests us as process analysts are precisely these relationships and rich data semantics. Our aim is to re-enrich these source data structures, transforming them into a **reality-close network structure**, retaining access to the context data of the source system while unlocking the full universally usable analysis potential of graph-based process structures.

Why hasn't this been done before?

We believe that there are two main reasons that the process mining industry remains Event Log Driven, even in the era of OCPM. First, the source systems processes are usually mined from are relational database driven. SQL-like databases drive somewhere in the region of **90%[16] of the software market**, and relational databases are relatively simple to convert to flat files. Coupled with the embedded ETL step, this makes the event log the easy answer to universal importing.

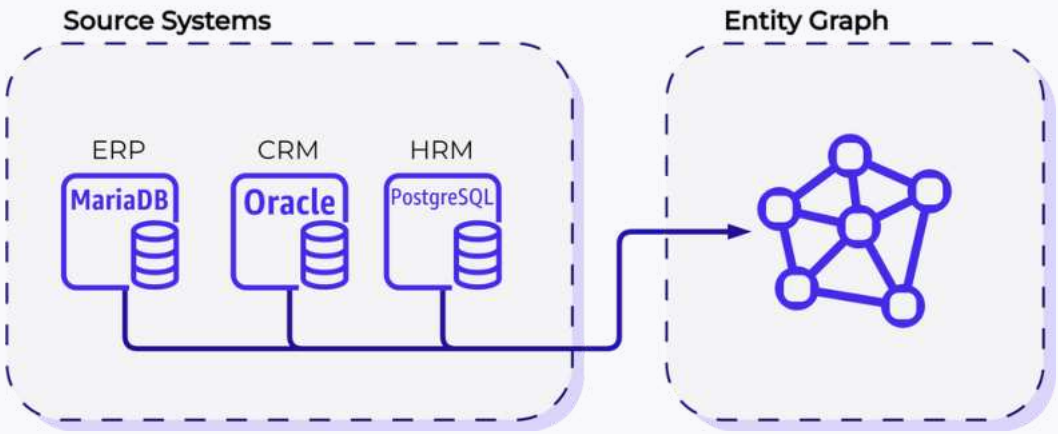
Second, the target systems – the process mining tools themselves - have all been built to handle event logs as an input, and it is extremely costly to change this in an existing tool, as it requires **rebuilding years of architecture** and re-onboarding clients that have invested significant capital into creating these event logs. Add on to this that large process mining providers have built-in event log converters for relatively standard systems like SAP, Salesforce, ServiceNow or Oracle[17] which cuts some of the cost out of initial setup. In this way, it becomes very difficult for an established player to justify the cost of making a major transformation in their approach.



Introducing Causal Process Mining

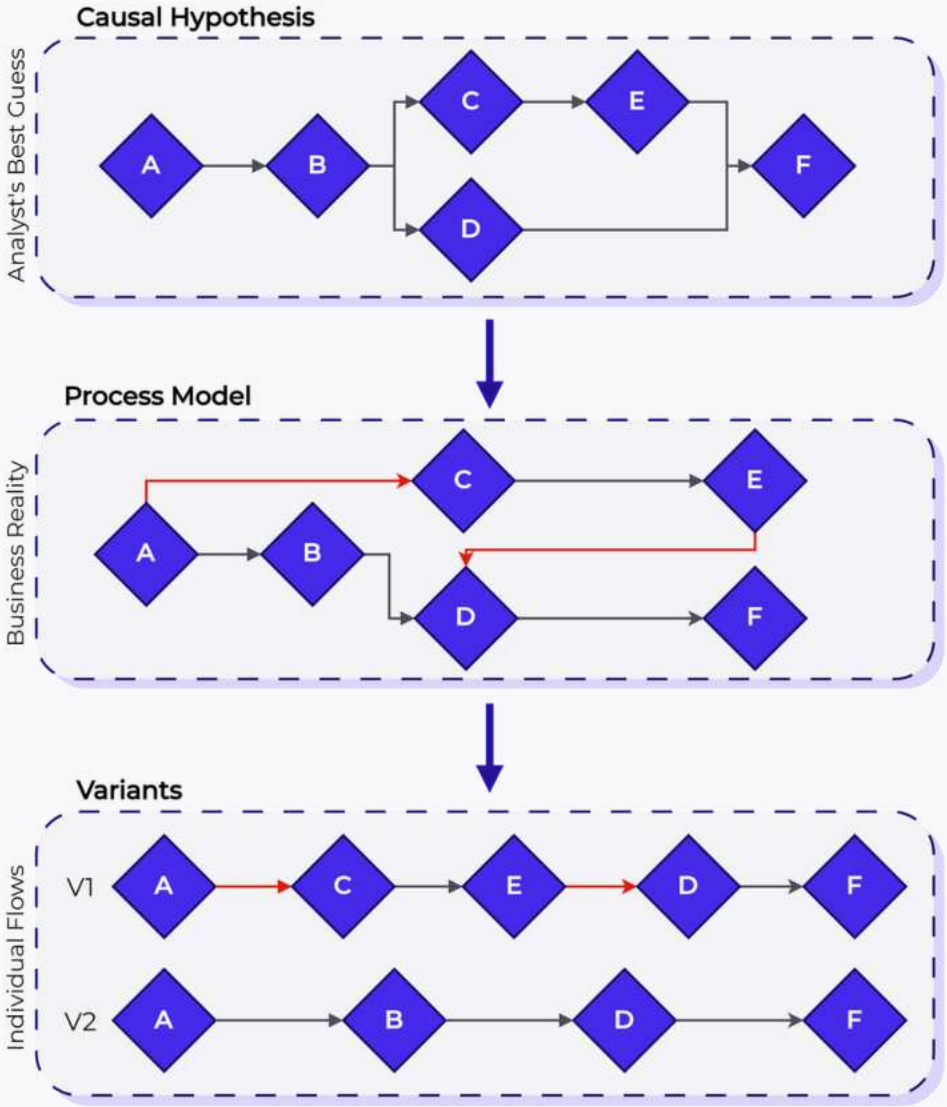
This is where we come in. Without the burdens of legacy implementation constraining us, we have built our solution from the ground up to create true-to-life, truly three-dimensional models of processes. Stemming from research done by our founder team[18], we call our approach **Causal Process Mining**, as it adds knowledge about causal dependencies to vision of object-centric process mining. We believe this to be true to the original vision for OCPM[19].

Causal Process Mining crawls business system databases and draws a network of connections between events occurring in the organization based on their digital traces, just like classical process mining. Unlike classical process mining, however, we do this by connecting directly to the source database and using relation mapping to pull a common thread through the entirety of the business process.



First, we **map every entity with its database key** to generate an unfiltered graph of the source database – we call this the **Entity Graph**. This is the hard part, and unfortunately, at this point, there is no good way of avoiding the need for database knowledge in process mining, even with our approach. After this point, however, we can use this graph to generate as many specific process models as we want, using primarily domain knowledge and a rapid prototyping philosophy. The assumption here is that every business has at least a few experts that have a rough idea of how their business processes are represented in the database – and that's all we need for an initial starting point.

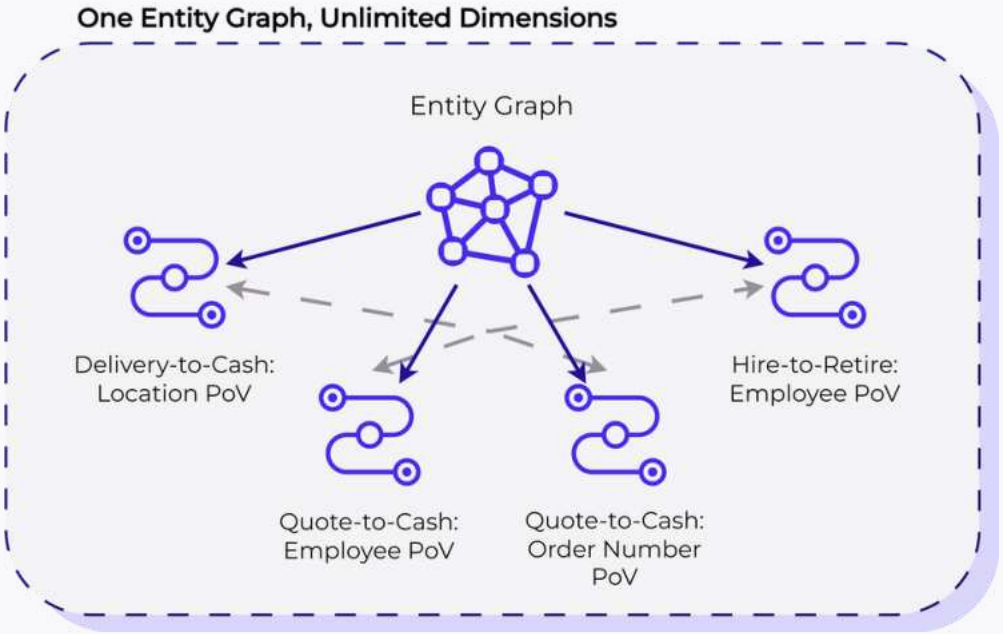
From here, the approach is simple – the analyst selects the business objects that are interesting to their analysis, maps these pre-existing connections into a **Causal Hypothesis** from the perspective of a specific unifying object type and lets the model run through the data and draw the real-world connections between the events. We call this a **Dimension**, as we can view the process graph from any number of dimensions by using a different **Case Identifier** – for example, we could trace the same process from the perspective of either an order ID, a location ID, or an employee ID. At any given point in this process, any context data such as financial figures, locations, employee IDs, or additional internal classifiers can be added to any event, on the fly, and used for later evaluation.



As we know both the expected process and the reality, the resulting dimension automatically differentiates between expected and unexpected process patterns. Further, the individual process Cases are aggregated at multiple abstraction levels, with individual traces or Instances being mapped to **Case Models** unified by the **Case Identifier**. These are aggregated into to **Variant Models** that represent the specific run pattern the Cases share, and finally into the overall **Process Model**, which unifies all events and transitions in the Dimension.

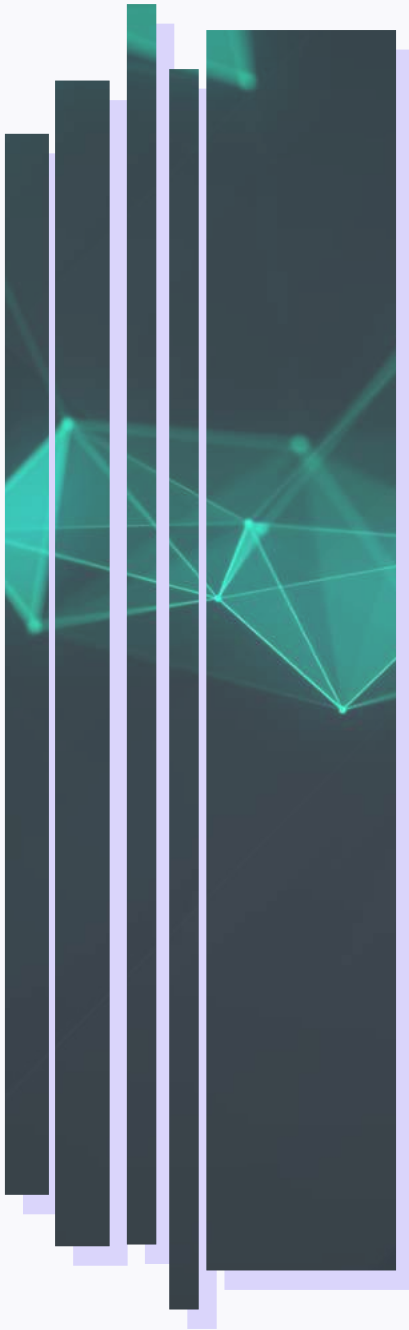
Using these tools, the analyst can go on to perform an unassisted discovery of their organization's process reality by iterating on or expanding their working dimension, or by creating new dimensions based on the already-generated process graph. All, in most cases, without the need for the assistance of the database administrator, process mining consultants, or someone from the strategic level of their organization at every step.

On top of this, since this approach generates new dimension on top of a single knowledge graph, there is no doubling of individual process events – each instance of a single Event A is shared between every Dimension deriving from the same Entity Graph B, opening completely new avenues for data science and AI applications.



Novel, but Grounded in Precedent

Most processes in a given organization are not comprised of simple one-to-one transitions – rather, each activity flows organically and can link to any number of entities and activities, both because they are highly complex and because they are often **based on human processes**. This last point is the most important – this makes real-world processes more akin to social networks than the classic timelog format that event logs emulate.



In 2009, Facebook migrated from Memcached MySQL tables to their now-famous TAO architecture[20], becoming one of the first big tech companies to enter the world of graph databases. Google followed suit in 2012 with the Google Knowledge Graph[21]. More recent, and perhaps even more relevant, was the large-scale shift in the logistics industry to using knowledge graphs during the COVID epidemic[22] to navigate extremely tough supply chain challenges.

We believe it is time for process mining to take this step as well. Instead of flattening event data even further, we think it's process mining's job to enrich it, drawing exactly this organic map of relationships between all the events that are represented in whatever systems organizations use. As a result, our approach allows us to not only represent business processes with more accuracy and detail, but it also unlocks **brand-new capabilities**, for which traditional process mining requires additional effort or simply does not have.

To name a few of these novel applications, this approach allows us to:

- Differentiate between **expected** and **unexpected** process transitions, identifying where the standard process was either not followed or where our assumptions about the process are wrong.
- Differentiate between **sequential** and **splitting** processes natively, as opposed to via compare-and-contrast analysis requiring multiple event logs.
- Draw accurate transitions between **instances** of events, instead of the industry standard all-to-all relationships necessitated by the event log approach.
- Map exact process transitions based on daisy-chained object identifiers, even if **no direct link** exists to the original object.
- View the same actual activities and objects from the **perspective of different IDs** on the fly, using these daisy-chains to fill the gaps in the data.
- Calculate a plethora of new benchmark process metrics, such as **process complexity** based on divergence or convergence, **adherence** statistics, and **concept drift** heuristics.

Where incumbent implementations of OCPM can be compared to the move from hierarchical to relational database structures in the 1970s[23], we believe the that the move that is truly needed at this point in the field's maturity is precisely the move that Facebook made in 2009 when they abandoned their initial relational databases in favour of their TAO model. Business processes are becoming **more linked and more complex**, and transparency and accountability more important as we move into an ever more digitalized future. This is why it is our solemn mission to drive the transformation of our industry into a direction that can keep up with these needs.

Applying the Theory
Use Cases and Advantages

There are some key advantages of this new approach that we touched upon in the previous chapter that we would like to explore in more detail here, especially in the context of how it improves upon the current standard approach. We would like to stress at this point that we consider this approach to still be in its infancy, and there is an astronomical solution space that it opens that is yet to be explored – we'll touch upon that in a later chapter of this paper. The applications we have highlighted here are those that we have already proven and validated with our initial partners and pilot projects, as well as the ones we find to be the most valuable at the current time as this approach embeds itself in the market and becomes more commonplace.

Optimising the Process Mining Process

We began this paper by discussing the financial and time costs of generating event logs for the classic process mining approach, especially in the context of multi-log OCPM. To recap – event logs are timely, costly endeavours because they require at least four highly qualified individuals to work together concurrently to generate and require large amounts of ETL right out of the gate.

With a direct-to-database approach and the pre-generation of the **Entity Graph**, the **Causal Process Mining** approach decouples the necessary interaction points in the initial implementation, as there is no need for a process mining expert. By using standardized knowledge sets and dropping the need to define which metadata are strategically important at the initial configuration of the graph, we drop the interaction need to only two people for the initial graph creation – a database administrator and a process expert.

The real gains start to realize themselves from this point on – the iterative nature of Dimension configuration mean less sunk costs and less decisions made up-front for the initial Process Model generation, and only incremental internal labour costs for successive dimensions based on the same Entity Graph. It is also fully plausible that this entire process is done completely hands-off, moving the value-add of external consultancy or internal process mining experts from a requirement for implementation to a force-multiplier for diagnosis and intervention.

All of this drops the implementation timeline of a **Causal Process Mining** tool to, at current maturity, **less than one month** including initial data preparation – which is already **half of the lower bound** found in Workfellow's market analysis. Every subsequent dimension configured on the same Event Graph costs a fraction of this time on top, and further development of the approach will improve these speeds further.

There is a strong case to be made that for organizations using standardized market-leading applications like Oracle or SAP's ERP systems, existing solutions on the market win out in this category because of existing standardized transformers that can stream source data without need for significant transformation effort. Over time, ERP systems tend to see an increase of their level of customization[24]. For a concrete example from the industry, Salesforce attributes a large part of their success in the market to being able to offer ad-hoc, on-the-fly customizations to its clients[25]. Adding to this are regional providers of enterprise software that are local champions in the SME space but are not large enough to have custom connectors built for them in existing solutions.

It seems to be a safe conclusion, then, that standard connectors are not a sufficient gap-closer for this metric, as at best, they only apply to a small subset of the market, and at worst, they will **stop working for existing customers** post-implementation and cause unexpected customization costs down the line. In addition to this, standard connectors are equally possible under our new approach as an accelerator to implementation, with more flexibility and less specific knowledge needs.

True Process Flows, without the Spaghetti

The elimination of the flattening problem and support for true many-to-many relationships between individual events in the **Entity Graph** allows the **Causal Process Mining** approach to represent business reality at a significantly higher granularity and removes the need for aggressive outlier management and on-purpose blind-spot creation, which we noted as being a common practice in the first section.

This has multiple benefits for process analysis. First and foremost, it allows 100% coverage of a dataset and makes it a purposeful choice to remove outliers, or, alternatively, to look at outliers alone in the process model, without compromising the analytical value of the resultant process graph.

Second, the cleaner nature of the depiction lowers the expertise floor required to understand the evaluations and use the graph to analyse business processes. Since a well-prepared **Causal Process Model** represents the true transitions per the business reality of the organization, we achieve a nearly-perfect fulfilment 2nd Nielsen Norman Usability Heuristic[26], which allows any user who has a working understanding of the process being evaluated to understand the process model with relative ease.

Finally, we open the possibility for new types of evaluations, such as identifying waiting times in batched processes, which is only possible thanks to **Causal Process Mining**'s ability to represent convergent and divergent process instances. For example, in a delivery of goods that need to be sourced from multiple suppliers, we can evaluate which suppliers delay the delivery of the whole package, and thereby can find out easily if split deliveries or better supplier management would help improve our customer's satisfaction.

Finding Out if the Beaten Path needs Guard Rails

The third major new use case unlocked with our method is true process adherence and quality assurance management. The possibility for many-to-many relationships between events and the existence of a **Causal Hypothesis** that is used to generate the **Process Model** enables both a visually and semantically clean separation of “expected” and “prohibited” process paths.

Additionally, the nature of the knowledge graph underlying the approach allows for a clean and simple taxonomy of the different kinds of violations present in business reality, and therefore a more granular filtering and analysis of the kinds of deviations present in a given process[27].

This functionality has myriad potential applications, such as:

- **Quality assurance** of enterprise software implementations, especially with heavy customization.
- **Delta analysis** for the change management needed to implement quality certifications such as ISO or SOC2.
- **Auditing of adherence** to processes necessary for these certifications or government regulation, if already employed, as risk management.
- **Generating supporting materials** for financial or regulations audits, reducing the cost burden of auditors to an organization with compulsory reporting.

Even outside of these regulation-centric use cases, this functionality allows an analyst to quickly and easily identify places where work is redone, done in the wrong order, or where necessary quality steps are circumvented – or, alternatively, find situations where there was a valid deviation from the prescribed process which creates efficiency or reduced complexity and can then be disseminated across the organization.

Complex Evaluations and Predictive Modelling

Finally, since the event knowledge is stored in a graph where every node is interconnected, there's near-endless potential for data science evaluations of process data prepared with Causal Process Mining. Since the analyst – or data scientist – can add any metadata from the source data at any point without changing the event log transformer, and can configure dimensions of the parent Entity Graph on the fly to create a network of causal links from the perspective of any core case identifier, it is possible to create a near-complete causal network of the entire organization – functionally, a digital twin of an organization's processes.

On the basis of this, it would be possible to create bespoke predictive or descriptive models that can inform the organization's decision-making, potentially helping to find problems before they arise. Additionally, this kind of network is prime for use as the base of an AI assistant for improving the organization's efficiency. Between these two approaches, some theoretical use-cases that become possible are:



- **A recommendation engine** for the highest impact/effort ratios problem locations in the organization.
- **A capital-maximizing model** for setting discounts for new clients based on segment, size, and location.
- **A best-practice distribution model** to determine how best to move your workforce to spread knowledge across the organization optimally.
- **A stock-purchase-prediction model** which informs the organization how much stock to buy during dips in supply price to fill demand until the next dip occurs.

While some of these use cases can be covered to some degree on the base of other existing tools and with traditional process mining, the nature of the core dataset being a knowledge graph makes it significantly more efficient and effective to use for these purposes, and expands the benefits of **Causal Process Mining** tools past simple process efficiency and adherence evaluations into the domain of general data-driven strategic governance.

Growing Pains

& Other Challenges

Of course, as with any fledgling, novel approach, **Causal Process Mining** does not come without its potential issues. In this section we'll briefly discuss some of these, as well as some mitigating factors that we see.

Environmental Challenges

- **Data Ownership -** A direct-to-database connection for the **Entity Graph** load naturally comes with more risks than a process with a decoupled, managed transformer in the middle. Without requiring a transformer, we omit a crucial opportunity for anonymizing customer data as a part of the core process, and therefore make it more likely that critical business data is represented in the process mining system without any form of scrambling. This requires cleaner data governance with client organizations and may require additional security controls in particularly sensitive cases.

- **Domain Expertise -** As with traditional process mining, **Causal Process Mining** has a relatively high need for domain knowledge in the implementing organization. This is currently not a problem that can be solved outright, but as already mentioned in the section on reality accuracy, we believe this to be mitigated by virtue of the approach being intrinsically more Usable from a systems design perspective[28].

- **Data Quality -** As mentioned before, without the Event Log transformation, Data Quality becomes a greater success factor in our approach when compared to the status quo. To create the **Entity Graph**, we require discrete data traces linking every event and object to be evaluated. In cases where this is not present, we need to default back to traditional-style ETL processes, manually creating these traces and re-adding the bias risk that we otherwise avoid with our approach. However, we believe that this was a much larger problem when the field of process mining came into being than it currently is, as data quality has been steadily increasing across most industries over the past two decades[29].

Performance and Technical Challenges

As we mentioned at the beginning of this paper, one other large advantage of traditional process mining is the computational efficiency of two-dimensional data structures. In contrast to this, **Causal Process Mining** is knowledge graph based, and thus is subject to exponential compute requirement growth[30]. The larger the organization's dataset, the more computational power is needed to process the causal network.

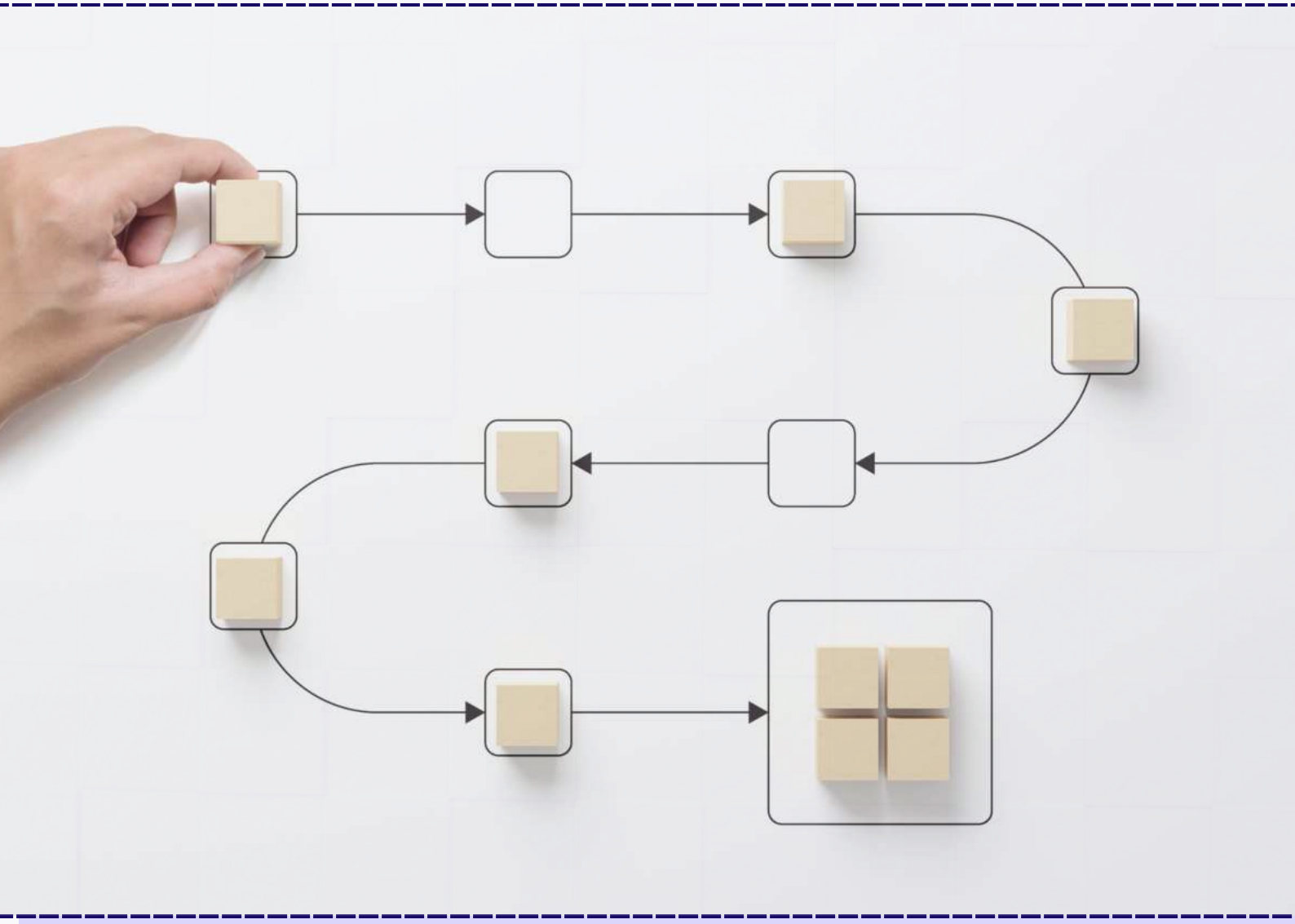
Dealing with this challenge requires the following measures:

- **Abstracting Process Model Generation from Event Graph Generation** – this approach lets us front-load more than half of the computational load for importing a dataset into a single initial step.

- **Delta Loading** – by adopting an approach of only loading the difference between the processed dataset and any changes to its configuration or contents, we can minimize the amount of additional compute needed for subsequent data updates.

- **Lazy-Indexing and Taxonomies** – by appending indices, hashes, and other abstracted identifiers to Known Entities such as Variants, and processing these in the background, we can increase the time-to-live of every iteration of a dimension while improving the load-performance of the graphs within the tool.

While research plays into our favour here by defining that efficient business process management approaches an efficacy limit from thirty events in a single process onwards[31], these challenges prevent us from bending that rule. We set this event limit on our dimensions both to improve the effectiveness of the tool's evaluations, as well as to avoid the problem of cubic scaling with both case and event counts.



The Steep Slope Up
Future Potential and Applications

Finally, we wanted to briefly touch upon the vast potential this approach offers in the future. While this section is primarily speculative and untested, we believe it is nonetheless important to highlight why we believe so strongly in **Causal Process Mining** as the future for this industry.

Neural Network Compatibility

It should come as no surprise that the most important point in this segment is the potential for **Causal Process Mining** for AI applications, just as it was no surprise that some of the largest players in the AI industry as those that implemented knowledge graphs early – Google, Facebook and Amazon. Knowledge graphs naturally mimic the form and function of neural networks[32], and the multitude of connections that are created in the process make them ideally suited for training graph neural networks[33].

We foresee great potential, which we already touched upon in the previous chapter, in predictive modelling based on this. The absolute *telos* of this concept would be a fully **organization-wide, multi-dimensional predictive approach to process mining**, with in-virtu 360-degree intervention simulation prior to organizational rollout, improving success rates and preventing costly initiatives from failing before they have been begun.



Automation Embedding

By creating a live parity between our Entity Graph and the source Database, Causal Process Mining opens many potential avenues into automated workflow embedding. Because there is no transformer in-between and all model adaptations can – in the future – be modified via API, it is feasible on the basis of this approach to create a fully **self-learning, self-optimizing and self-healing automated workflow**.

Coupled with an appropriate AI Model, **Causal Process Mining** could pave the way to self-managing production lines and smart, organization-spanning enterprise software systems that not only react to but predict what the needs of the larger organization are and adjust themselves appropriately.

The One-Stop Shop Data Lake

As we mentioned in earlier chapters, one of the major limitations of existing approaches is the single-task nature of the resultant data structures. By leveraging a theoretically infinitely semantically scaling **Entity Graph**, process mining ceases to be a technological endpoint in the organization's software ecosystem. Instead, we can position process mining at the centre of the data landscape, using its capabilities to create a digital twin not just of our organization's processes, but of every entity in our organization and its interrelationships.

The classic use-case of process models and visual analytics at this point is no longer the single central point of interaction, but just one of a myriad of analytical approaches that can be used to optimize the business, for profitability, for employee well-being, for social responsibility – for any goal the organization chooses. The data lake created by process mining then begins to resemble its namesake: a central point around which to settle and build out from, a central source for **universal organizational intelligence.**

Conclusion

A Thank You and a Call to Join Us

In this paper, we discussed the problems with traditional process mining that we set out to solve with our new approach – the cost, in both money and time, the loss of detail in the representation, and the limited uses of the resulting data structures.

We then proposed **Causal Process Mining** as a way to addresses these problems, as well as open completely new use-cases and value propositions by using graph databases as the central data structure and embedding the Causal Model into the implementation and discovery flow.

Finally, we discussed the challenges that we currently face, some of which are also faced by our predecessors – domain knowledge, data ownership and data quality. We believe these challenges to be comfortably – if not easily – surmountable in the foreseeable future and expect them to act as stepping stones towards the bright, transparent future that this approach unlocks.

In short – we firmly believe that this is the direction process mining needs to go as an industry from here on out, and **it is our mission to drive this change**. Two-dimensional data structures are not representative of the real world and limit us in ways that are as unnecessary as they are unfortunate. However, we recognize that the work that has been done so far has been necessary and should not be seen as a regret, but rather an achievement.

In that spirit, while we are the first on the scene for this new approach to managing process discovery, we look forward to welcoming both our older and younger counterparts from the industry to join us in ushering in **a new, brilliant world of organizational transparency and decision-making**.

And because no future oriented white paper is complete without a cliché yet powerful quote, we will end on this one from Satya Nadella, CEO of Microsoft:

> *"The future is not about competition, but collaboration. The future belongs to those who are willing to sacrifice their ego to a more humane way of living. Collaborate with others rather than compete."*



Authors



Evgeny Zhdanov, MBA

User-centric design obsessive by nature, product manager by trade, Evgeny leads Noreja's product team as Head of Product. With a background in digitalization and agile transformation consulting, he draws on his broad industry experience to build understanding and solutions for our users' needs.

Dr. Lukas Pfahlsberger



Lukas built his career prior to Noreja building experience at Ernst & Young, consulting for large multinationals, while simultaneously pursuing a doctorate in Process Management. As CEO of Noreja, he drives strategy, operations and sales while managing partner and academic ecosystem outreach.



Prof. Dr. Jan Mendling

Jan - Einstein Professor at the HU Berlin, co-author of Business Management textbooks used at over 250 universities worldwide, co-founder of the Berlin BPM Offensive, member of the IEEE Task Force on Process Mining - the list goes on. Jan lives and breathes Process Management and Mining.

Sources

1. PYMNTS. (2015, July 2). The history of PayPal: Important company dates. PYMNTS.Com. https://www.pymnts.com/in-depth/2015/throwback-thursday-paypals-biggest-days-in-history/
2. Kemp, S. (2024, January 31). Digital 2024: Global overview report — DataReportal – global digital insights. DataReportal – Global Digital Insights. https://datareportal.com/reports/digital-2024-global-overview-report
3. van der Aalst, W., Weijters, T., & Maruster, L. (2004). Workflow mining: Discovering process models from event logs. IEEE Transactions on Knowledge and Data Engineering, 16(9), 1128–1142. https://doi.org/10.1109/tkde.2004.47
4. V van der Aalst, W. M. P. (2019). A practitioner's guide to process mining: Limitations of the directly-follows graph. Procedia Computer Science, 164, 321–328. https://doi.org/10.1016/j.procs.2019.12.189
5. What is an Event Log. (n.d.). Celonis. Retrieved April 11, 2024, from https://academy.celonis.com/learn/video/what-is-an-event-log
6. Kampik, T., & Weske, M. (2022). Event log generation: An industry perspective. In Enterprise, Business-Process and Information Systems Modeling (pp. 123–136). Springer International Publishing. http://dx.doi.org/10.1007/978-3-031-07475-2_9
7. Numminen, L. (2024, January 8). How much does process mining cost? 2024 pricing guide. https://www.workfellow.ai/learn/how-much-does-process-mining-cost
8. Cost of change. (n.d.). Praxis Framework. Retrieved April 11, 2024, from https://www.praxisframework.org/en/library/cost-of-change
9. van der Aalst, W. (2016). Analyzing "spaghetti processes." In Process Mining (pp. 411–427). Springer Berlin Heidelberg. http://dx.doi.org/10.1007/978-3-662-49851-4_14
10. Tax, N., Sidorova, N., & van der Aalst, W. M. P. (2018). Discovering more precise process models from event logs by filtering out chaotic activities. Journal of Intelligent Information Systems, 52(1), 107–139. https://doi.org/10.1007/s10844-018-0507-6
11. Suriadi, S., Andrews, R., ter Hofstede, A. H. M., & Wynn, M. T. (2017). Event log imperfection patterns for process mining: Towards a systematic approach to cleaning event logs. Information Systems, 64, 132–150. https://doi.org/10.1016/j.is.2016.07.011
12. van der Aalst, W. M. P. (2023). Object-Centric process mining: Unraveling the fabric of real processes. Mathematics, 11(12), 2691. https://doi.org/10.3390/math11122691

13. van der Aalst, W. M. P., & Berti, A. (2020). Discovering object-centric petri nets. Fundamenta Informaticae, 175(1–4), 1–40. https://doi.org/10.3233/fi-2020-1946
14. Detwiler, B. (2023, January 16). What is object-centric process mining? Celonis. https://www.celonis.com/blog/what-is-object-centric-process-mining-ocpm/
15. Graph vs relational databases - Difference between databases - AWS. (n.d.). Amazon Web Services, Inc. Retrieved April 11, 2024, from https://aws.amazon.com/compare/the-difference-between-graph-and-relational-database
16. DB-Engines ranking. (n.d.). DB-Engines. Retrieved April 11, 2024, from https://db-engines.com/en/ranking
17. Tools - Process mining software comparison. (n.d.). Process Mining Software Comparison -. Retrieved April 11, 2024, from https://www.processmining-software.com/tools/
18. Waibel, P., Pfahlsberger, L., Revoredo, K., & Mendling, J. (2023). Causal Process Mining from Relational Databases with Domain Knowledge.
19. van der Aalst, W. M. P. (2019). Object-Centric process mining: Dealing with divergence and convergence in event data. In Software Engineering and Formal Methods (pp. 3–25). Springer International Publishing. http://dx.doi.org/10.1007/978-3-030-30446-1_1
20. Marchukov, M. (2013, June 25). TAO: The power of the graph. Engineering at Meta. https://engineering.fb.com/2013/06/25/core-infra/tao-the-power-of-the-graph/
21. Singhal, A. (2012, May 16). Introducing the Knowledge Graph: Things, not strings. Google. https://blog.google/products/search/introducing-knowledge-graph-things-not/
22. Zimmerman, A. (2021, December 14). How knowledge graphs have transformed the supply chain industry during COVID. Graph Database & Analytics. https://neo4j.com/blog/how-knowledge-graphs-have-transformed-the-supply-chain-industry-during-covid/
23. National Research Council, Computer Science and Telecommunications Board, & Committee on Innovations in Computing and Communications: Lessons from History. (1999). Funding a revolution: Government support for computing research. National Academies Press.
24. Ondrej, Z., & Munkvold, B. E. (2011, July 9). ERP system customization in smes: A multiple case study. PACIS 2011 Proceedings. https://objects.scraper.bibcitation.com/user-pdfs/2024-04-11/67219c5b-860d-4748-bdd5-cc6ea08faf6e.pdf
25. Spiegel, S. (2024, March 8). Salesforce's First 25 Years: Blazing a Trail to the #1 AI CRM. Salesforce. https://www.salesforce.com/news/stories/25th-birthday-feature/
26. Nielsen, J. (1994, April 24). 10 usability heuristics for user interface design. Nielsen Norman Group. https://www.nngroup.com/articles/ten-usability-heuristics/

27. Pfahlsberger, L., Rubensson, C., Knoblich, S., Vigdof, M., & Mendling, J. (2023, November 28). Multi-perspective path semantics in process mining based on causal process knowledge. Companion Proceedings of the 16th IFIP WG 8.1 Working Conference on the Practice of Enterprise Modeling and the 13th Enterprise Design and Engineering Working Conference. https://objects.scraper.bibcitation.com/user-pdfs/2024-04-11/7f411af4-6366-417f-9d2c-dd4405dbb4bb.pdf
28. Gulliksen, J. (n.d.). Designing for usability. Retrieved April 11, 2024, from https://www.it.uu.se/research/hci/publications/papers/diss189/summary.html
29. Khong, I., Aprila Yusuf, N., Nuriman, A., & Bayu Yadila, A. (2023). Exploring the impact of data quality on decision-making processes in information intensive organizations. APTISI Transactions on Management (ATM), 7(3), 253–260. https://doi.org/10.33050/atm.v7i3.2138
30. Thompson, N., Manso, G., & Ge, S. (2023). The importance of (exponentially more) computing power. Academy of Management Proceedings, 2023(1). https://doi.org/10.5465/amproc.2023.365bp
31. Mendling, J., Reijers, H. A., & van der Aalst, W. M. P. (2010). Seven process modeling guidelines (7PMG). Information and Software Technology, 52(2), 127–136. https://doi.org/10.1016/j.infsof.2009.08.004
32. Wu, L., Cui, P., Pei, J., & Zhao, L. (2022). Graph neural networks: Foundations, frontiers, and applications. Springer Nature.
33. Scarselli, F., Gori, M., Tsoi, A. C., Hagenbuchner, M., & Monfardini, G. (2009). The Graph Neural Network Model. IEEE Transactions on Neural Networks, 20(1), 61–80. https://doi.org/10.1109/TNN.2008.2005605