A Digital Process Twin Conceptual Architecture for What-If Process Analysis

Ivan Compagnucci[1] [0000−0002−1991−0579], Barbara Re[2] [0000−0001−5374−2364], Estefanía Serral Asensio[3] [0000−0001−7579−910X], and Monique Snoeck[3] [0000−0002−3824−3214]

[1] Gran Sasso Science Institute, Italy
{name.surname}@gssi.it
[2] University of Camerino, Computer Science Division, Italy
{name.surname}@unicam.it
[3] KU Leuven, Faculty of Economics and Business, Belgium
{name.surname}@kuleuven.be

Abstract. Business processes require continuous changes or interventions to remain efficient and competitive over time. However, implementing these changes—such as reordering or adding new tasks— can negatively affect the overall process performance. A longstanding problem in Business Process Management is that of forecasting *ex-ante* the values that process performance measures will assume after implementing changes. To achieve this, the concept of Digital Process Twins, which extends the well-established Digital Twin paradigm, paves the way for new interesting opportunities. Digital Process Twins enable enhanced *what-if* analysis by virtually predicting process performance under various changes, thus allowing for informed decision-making before actuating process changes in the real world. However, despite recognition as one of the new key enablers of modern process re-engineerization, a comprehensive approach to implementing Digital Process Twins is still lacking. This paper proposes a novel conceptual architecture for deploying Digital Process Twins to address this gap. Additionally, we introduce Dolly, a framework that implements such conceptual architecture using a multi-modeling approach combining domain data and process modeling along with a data-driven process simulation technique.

Keywords: Business Process · Digital Process Twin · Internet of Things

1 Introduction

Nowadays, organizations constantly strive to enhance and sustain the efficiency and performance of their operational processes [16]. This necessity is fueled by several factors, including the increasing competitiveness of the global market, environmental shifts, variations in resource availability, emergent business opportunities, and the advent of new technologies [15]. A notable example of these advancements is the emerging field of IoT-Enhanced Business Processes [8,9,33], where IoT devices are increasingly being integrated into processes to optimize further and automate business operations. However, for a long time, a problem in the field of Business Process Management is that of *what-if process analysis*: predicting the values that one or more process performance measures will assume after a given business process changes or interventions [4, 15, 18].

A similar problem has been addressed in mechanical and industrial engineering using the Digital Twin paradigm. Digital Twins are virtual replicas of real-world systems synchronized at specific levels of detail. They accurately predict the performance and behavior of their physical counterparts over time, offering valuable insights for optimization and decision-making [19]. Initially adopted in the manufacturing sector to virtually replicate, simulate, and predict the performance of physical machines, the concept of Digital Twin is starting to be applied to organizational processes, providing a new approach to re-engineering modern business processes [15, 18]. Gartner estimates that by 2026, 25% of global enterprises will move towards creating Digital Twins for their business processes [22].

In light of this, the integration of Business Process Management practices with the Digital Twin paradigm is being seen as a promising solution for helping organizations manage process changes while maintaining resilience and control over their operations [4, 15, 18]. Just as traditional Digital Twins replicate and predict the performance of physical assets, Digital Process Twins offer analogous capabilities for business processes. Implementing changes in business processes typically involves significant time, resources and risk of failure, leading to high expenses [16]. This integration facilitates *what-if* process analysis, allowing organizations to simulate potential changes and predict their impact on process performance *ex-ante* in a virtual, safe, and risk-free environment [15, 18, 25]. However, despite being recently recognized as a key enabler for digital transformation in organizational processes [4, 15, 25], there is currently no detailed framework for fully exploiting the opportunities that a Digital Process Twin can provide [18].

The contributions of this work are twofold. First, we propose a novel conceptual architecture for implementing Digital Process Twins. The proposed architecture employs heterogeneous digital models and Business Process Management techniques to replicate the *as-is* process and reason about the performance of a *to-be* process after virtually implementing process changes. Secondly, we present Dolly, a framework that implements the proposed Digital Process Twin's conceptual architecture. Dolly uses a multi-modeling approach that combines an IoT domain model with the MERODE methodology and BPMN, enabling the simulation and prediction of process changes' impact on performance before real-world implementation.

The rest of the paper is organized as follows. Section 2 presents background knowledge. Section 3 presents a conceptual architecture for implementing Digital Process Twins. Section 4 introduces the Dolly framework supporting the Digital Process Twin conceptual architecture in practice. Section 5 reports on the Dolly evaluation. Finally, Section 6 discusses related works, and Section 7 summarizes and concludes the paper.

2 Background

This section overviews the most relevant aspects of deploying the Digital Process Twin. First, we introduce MERODE, a model-driven method used to support the design of digital models for Digital Process Twin. Additionally, we discuss the Business Process Simulation technique, which is fundamental for conducting "what-if" analyses and estimating business process performance.

2.1 The MERODE Methodology

Adopting a Model-Driven Engineering approach in developing Digital Twins is fundamental to fully leverage their potential [19, 24]. A noteworthy approach within this domain is the MERODE methodology [31]. MERODE uses object-oriented domain modeling to develop enterprise information systems, structuring the design and implementation of intra-organizational enterprise information systems into three distinct layers: the Domain layer, the Information System Services layer, and the Business Process layer [31].

The Domain layer defines *business objects*, including their attributes and relationships. A business object represents a real-world entity relevant to a business process, such as data, documents, people, events, or other elements participating in a business process [16]. Examples of business objects could include *Container* and *Shipment*, which can be instantiated to link a container with a specific shipment. Additionally, a *Sensor* equipped on each container constantly monitors and tracks data in real-time, providing comprehensive information about the shipment's status and conditions. The domain layer enables code generation from a conceptual model named "MERODE Domain Model", facilitating the transition to a functional prototype of the information system [30, 31]. The MERODE Domain Model consists of three views: a *Class Diagram*, an *Object Event Table*, and a set of *Finite State Machines*. The class diagram defines business objects and their relationships, while the object event table maps event types triggered by business objects. When an event fires, it triggers the execution of methods on business objects used to create, modify, or end business object instances. Finite state machines specify the life cycles of business objects, depicting object behavior triggered by events. A MERODE Domain model can be modeled using the MERLIN Modeling Tool[4], providing model consistency and correctness assessment features.

The Information System Services layer acts as a bridge between business objects and business processes. Input services update the business objects by modifying their attributes or state, while output services provide access to data.

The Business Process layer sits above the Information System Services layer. Its purpose is to facilitate interactions between processes and the Domain layer via Information System Services, ensuring the update and exchange of information with business objects.

[4] https://www.merlin-academic.com/

2.2 Data-Driven Business Process Simulation

Traditional Business Process Simulations allow business experts to estimate the performance of business processes under varying conditions and constraints [1]. To run a simulation, a Simulation Model is required. This model digitally replicates real-world processes, including detailed mappings of process flows, activities, decision points, and resources. In addition, it necessitates a set of Simulation Parameters that represent quantitative variables such as activity processing times and costs used to ensure that the Simulation Model accurately reflects real-world conditions [1, 28]. However, the manual creation and fine-tuning of Business Simulation Models is an error-prone task, involving a complex set of models and parameters defined and assessed manually by business experts. This approach often leads to inaccurate models and requires significant time to identify the optimal scenario for desired performance outcomes [6, 14].

Data-driven process simulation offers a solution by leveraging real data to discover accurate and enhanced simulation-ready models [6, 14]. Unlike traditional process simulations, which rely on manually gathered and interpreted information, data-driven simulations utilize historical and real-time data from event logs. Mining techniques based on past event logs of the process [6, 23] are employed to ensure that simulation-ready models and parameters are reasonable and aligned with reality [1]. Historical data provide retrospective insights through process mining techniques, while real-time data enable continuous updates to the simulation model, ensuring it accurately reflects the current state of the process during the simulation [14].

Once the simulation model is configured, it is ready to be simulated, and results can be interpreted. To this end, Key Performance Indicators (KPIs) are crucial for evaluating the performance and effectiveness of business processes. KPIs are values for measuring the effectiveness in achieving specific goals of a business process [1]. They include metrics such as cycle time distribution, waiting time distribution, cost distribution, and resource utilization, providing benchmarks for evaluating overall process performance. By assessing the KPIs, the *what-if* questions mentioned above can be answered, and different process redesigns can be compared.

3 Conceptualizing Digital Twins

Implementing a Digital Twin infrastructure is a non-trivial task [19, 26]. Despite the emergence of various implementations from both research and practical applications [17, 26], no single solution can be considered a silver bullet for implementing a full-fledged Digital Twin [19]. A Digital Twin environment typically includes a collection of interconnected models and data that replicate a real-world system [19]. It provides services, including design, development, analysis, simulation, and optimization, enabling a thorough understanding and enhancement of the replicated system's performance [17, 19].

3.1 A Conceptual Architecture for Digital Twin

In [17], the authors explored various characterizations of the core elements of Digital Twins. This effort was directed at providing a clearer understanding of the foundational components of Digital Twins. They proposed a generic and conceptual architecture for facilitating the systematic engineering of new domain-independent Digital Twin applications. According to [17], a Digital Twin adheres to a three-component architecture described as a three-element tuple:

$$A_{DT} = \langle \textit{Actual System}, Models, Data \rangle$$

Where the *Actual System* represent a real-world system or object; *Models* provide digital representations of the Actual System; and *Data* represents current and historical data of the Actual System, crucial for instantiating digital models. The three main components of the architecture are described in the follows.

- The **Actual System** refers to the real-world system that the Digital Twin aims to replicate. It involves collecting, storing, calculating, and inferring data specific to the system. These activities are essential for the Digital Twin to capture relevant aspects, features, and relationships of the Actual System within its operational contexts and environments.
- The **Data** component is about storing and representing current and historical data from the Actual System. Data and information are important to accurately provides information to models and reflect the actual system in the digital space of the Digital Twin environment, enabling accurate and fair analysis.
- The **Models** establish digital representations of the Actual System considering different perspectives. As stated by [17], it includes three types of models: *descriptive*, *predictive*, and *prescriptive*. *Descriptive models* capture and organize data to accurately replicate the Actual System. *Predictive models* support decision-making using aggregated data and insights from descriptive models to anticipate future system behavior and conduct “what-if” analyses. Finally, *Prescriptive models* incorporate insights from “what-if” analyses into adaptive actions aimed at optimizing the Actual System.

3.2 A Conceptual Architecture for Digital Process Twin

Digital Process Twins have recently been acknowledged as crucial enablers for digital transformation within organizational processes [4, 15, 18]. However, there is still a lack of comprehensive implementations to effectively leverage the Digital Process Twins paradigm. In the following, we propose a conceptual architectural approach tailored for the engineering and deployment of Digital Process Twins, drawing upon the conceptualization outlined in the previous Subsection. Figure 1 depicts a visual representation of the proposed conceptual architecture.

The **Actual System** here refers to a business process, which consists of actions, events, and decisions that lead to creating a service or product [16]. Typically, business processes encompass various perspectives, which the authors in [27] categorize into six distinct perspectives described in the following.

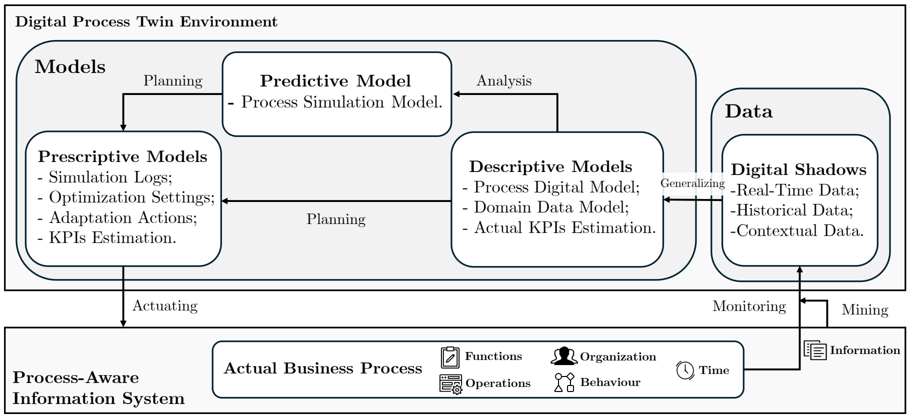


Fig. 1: The Conceptual Architecture for Digital Process Twin.

- **Function Perspective:** atomic activities representing specific business tasks within the process.
- **Behavior Perspective:** dynamic behavior including control flow, activity order, and constraints.
- **Information Perspective:** data used/generated in the process, organized via domain models (class diagrams, finite state machines).
- **Organization Perspective:** roles of participants and organizational units, ensuring proper task assignment.
- **Operation Perspective:** implementation details and integration with application services, supporting business functions.
- **Time Perspective:** temporal constraints like deadlines and durations, ensuring timely execution.

Process-Aware Information Systems integrate and manage these business processes by incorporating the aforementioned perspectives, facilitating control, monitoring, and analysis [27]. The information generated by these systems provide valuable information and data on various aspects of the process. These systems generate valuable information, including historical data stored in event logs and real-time data on ongoing process instances, which offer essential information for creating a digital process replica.

The **Data** component entails collecting and storing process-relevant data directly from the process and the Process-Aware Information System. Data are organized through *Digital Shadows*, which are abstracted and aggregated data structures that provide a one-way data flow from the Actual System to its digital representation [3, 25]. Information is transmitted to the Digital Shadow to establish a synchronous linkage between the Actual System and its corresponding Digital Process Twin. Data is fundamental for two reasons: first, it instantiates digital models that accurately replicate the Actual System; second, by populating these models, it enables detailed analyses that provide insights and drive improvements in the Actual System. The data flow, represented in Figure 1 by the "Monitoring/Mining" arrow, illustrates two methods of data collection: real-time monitoring of ongoing business process instances and historical data extraction using Process Mining techniques [2]. Real-time data includes information about the current status of the process (i.e., resource usage, active tasks, actual process KPIs). Historical data, including event logs of past process executions, organizational documents outlining procedures, and additional contextual data, provides valuable information for Process Mining analyses [2].

Considering the **Model** component, *descriptive models* aim to create a digital replica of the process [17]. Therefore, the first step is defining a model able to properly represent the actual business process embedding the six typical perspectives of business processes described below. Ensuring the quality of the process model is crucial, because it enables precise monitoring, analysis, and optimization of business processes, leading to improved efficiency, predictive maintenance, and informed decision-making [16, 35]. In this context, Business Process Model and Notation (BPMN) stands out as the most common and effective standard for designing a business process model for organizations [10, 11]. A BPMN diagram details the sequence of activities, control rules, and interactions between process participants, providing a clear and comprehensive representation of the entire process. It represents specific *behaviors*, *functions*, *operations*, *organizational* and *time* perspectives of the process. The model of the process is obtained by adopting process mining discovery algorithms [2], which analyze event logs from the Process-Aware Information System to ensure the model is accurate and reflects reality. In parallel, the domain data model manages the *information* perspective, organizing and structuring data relevant to the process. This includes class diagrams and finite state machines that define the relationships and states of business objects, ensuring data integrity and supporting the retrieval of process-related information.

To conduct *what-if* analyses, a *predictive model* representing the digital replica of the actual business process is employed. However, to implement and test new process changes, business experts must manually adjust the process structure (i.e., reordering tasks and adding new resources). For this reason, the digital replica should be modified by (i) manually implementing the necessary changes to the process model; (ii) discovering optimal *Simulation Parameters* using existing mining approaches on historical data [6, 23]; (iii) leveraging real-time data from a Domain Model [27]. This enables the creation of a data-driven process simulation model, allowing for the virtual implementation of changes and the estimation of the new process's performance through simulation. Finally, simulation insights can be translated into the form of *Prescriptive Models*. They consist of estimating KPIs and analyzing event logs to reason about the impact of changes made to the process. These insights are translated into actions, evaluated by business experts, and, if beneficial, implemented in the actual business process. To complete the feedback loop between the Digital Twin and the Actual System, the "Actuating" arrow involves implementing and executing actions on the Actual System based on prescriptive models. This approach helps reduce costs, save time, and provide a risk-free environment for virtual testing.

4 DOLLY: A Framework for Implementing Digital Process Twins

This section introduces DOLLY, a framework based on the Digital Process Twin conceptual architecture proposed in Subsection 3.2 It adopts a multi-modeling approach, integrating domain data models formalized with the MERODE methodology, the standard BPMN language for process modeling, and data-driven simulation techniques for what-if process analysis. Figure 2 provides an overview of DOLLY, highlighting its three key components: the Actual System, Data, and Models.

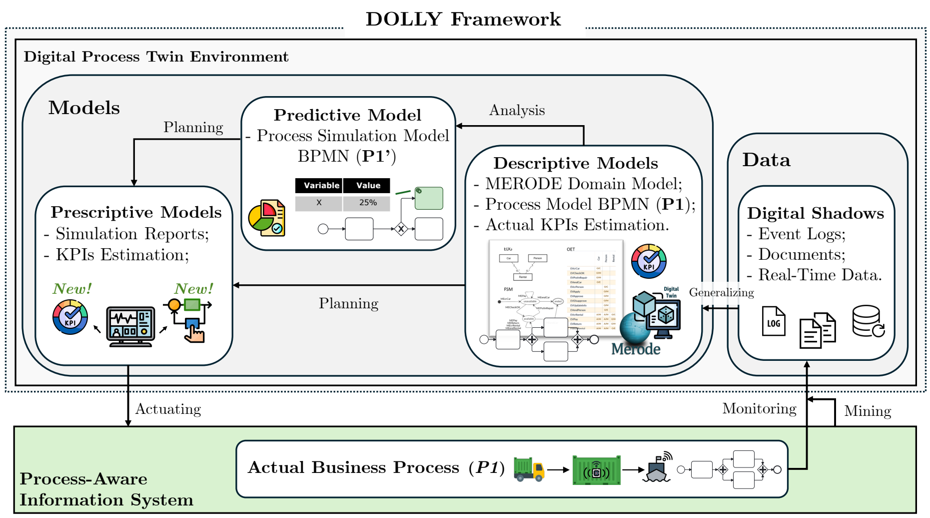


Fig. 2: DOLLY: Framework Overview.

To create a digital copy of the **Actual Process** *(P1)*, DOLLY allows to leverage data from the Process-Aware Information System that implements the actual business process. The **Data** extracted includes real-time information from ongoing process instances via an embedded Camunda Engine, as well as historical data obtained by uploading an event log representing previous process executions. The event log is used to discover the structure of the actual BPMN process model *(P1)* through Process Mining techniques [2]. On the other hand, Real-time data are used to instantiate the MERODE Domain Model. In [32], the authors demonstrate how MERODE bridges the gap between data and process modeling by linking these two domains formally. It allows the handling of domain process data by continuously monitoring business objects' status, relationships, actions, and actual process KPIs. This enables real-time management of their data, providing current status information within the process. Moreover, MERODE supports formal verification, reusability, and flexibility [31], creating descriptive models that reflect business processes from multiple perspectives.

Then, to evaluate the impact of potential process changes, a new BPMN model (*P1'*) is derived by modifying the digital counterpart *P1* of the actual process. Unlike *P1*, the *P1'* model necessitate additional features. First, it includes manual changes applied by business experts, implementing the desired changes to the process. Additionally, to effectively run simulations, *P1'* requires defining simulation parameters. These parameters are discovered using SimuBridge [23], which allows mining techniques to be performed on historical process data, ensuring that simulations are based on real information. Furthermore, *P1'* integrates real-time data from the MERODE Domain Model, aligning domain data with the current state of the ongoing process instance. This real-time data is essential for maintaining the accuracy and relevance of the simulations. By simulating *P1'*, which acts as a *predictive model*, it is possible to conduct "what-if" analysis within the Digital Process Twin, allowing for the evaluation of potential changes and providing valuable insights into their impact before real-world implementation. To run simulations on BPMN models, Dolly embedded BIMP UI, a business process simulator. This integration allows users to simulate business processes effectively, leveraging a user-friendly interface to visualize, analyze, and download simulation results.

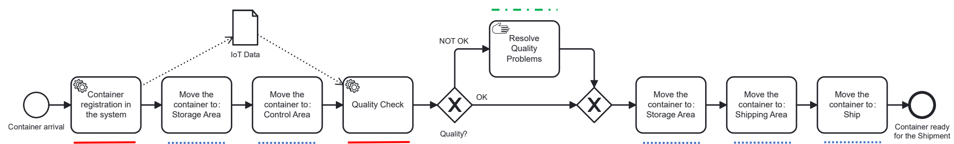
Business experts then evaluate the impact of the changes on process performance by carefully analyzing the simulation results. If the performance improves or remains unchanged, the changes suggested by the prescriptive models can be considered for implementation. If performance does not improve, *P1'* is revised and tested again. This iterative approach enables continuous process adjustments based on real-time data, simulation feedback, and desired process changes.

5 Framework Evaluation

This section presents a real-world implementation of Dolly in a smart harbor scenario. The objective is to evaluate the framework's capabilities. A smart harbor represents a technologically advanced port that leverages innovative technologies and data-driven solutions to enhance operational efficiency [29]. This scenario focuses on an IoT-enabled business process that represents modern processes designed for automation through IoT integration. While smart harbors encompass various processes, we will focus on container dispatching.

5.1 The Container Dispatch Business Process

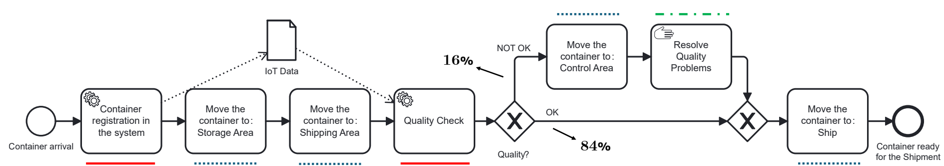
Context. The process starts with the arrival of a cargo container at the harbor and ends when it is loaded onto a cargo ship, indicating that it's ready for shipment. The containers involved in dispatching are equipped with IoT devices (i.e., RFID sensors) to track their status during the dispatching. When containers reach the harbor, their information (i.e., IoT data and shipping documents) is recorded in the system and transmitted to the Storage Area. Then, it is relocated to the Control Area for quality inspection. Quality control is conducted by cross-referencing the container's arrival data with the information gathered during manual quality inspection. If the container fails the quality test, a manual inspection is conducted to address potential quality issues, and the container is then returned to the Storage Area. Once quality problems are resolved, the container is moved to the Shipping Area and loaded onto the cargo ship.



(a) Discovered Actual Business Process (*P1*).

Costs:	Activities Duration:	Scenario Specification:
Worker 1: 20€/hr – 24/7	Moving the Container: 25/35mins – Uniform	Total N°of Instances: 500 Containers
Worker 2: 30€/hr – 24/7	System Registration: 5 mins – Fixed	Instances Arrival Time: 1 Container/hr
Worker 3: 25€/hr – 24/7	Solving Quality Problems: 1 hour - Fixed	Work schedules: 24/7

(b) Mined Simulation Parameters.



(c) Adapted Scenario Implementing Changes (*P1'*).

Fig. 3: Simulation Parameters and Models of the Container Dispatch Process.

Process Data. The event logs were generated considering two primary data sources: (i) logs from the smart harbor system, which contains the sequences of activities performed for each process instance and trace attributes, and (ii) IoT sensor data, which tracks the arrival and quality of the containers considering temperature and humidity. The process event logs were generated using CDLG [20], a tool specifically designed to create synthetic event logs integrating concept drifts and noise, such as missing event data. Event logs are based on patterns and data observed in real-life operations from the Tuscan Port Community System [5]. The event log comprises 7 activities, 3 resources, 33,910 events related to 5000 cases, and 67 execution paths (process variants).

[5] https:/tpcs.tpcs.eu/

5.2 Use Case Instantiation

Actual Business Process. As a first step, we employed the inductive miner process mining algorithm to discover the actual business process *P1* from the event log, applying a threshold to mitigate noise. The process structure has been slightly adapted to better align with real-world operations. Figure 3a depicts the process model discovered (*P1*). Automated tasks are designated as service tasks, while the manual task of resolving quality problems is identified as a manual task. Additionally, a data object has been incorporated to represent IoT data utilized during container registration and quality check activities. The initial container registration in the system involves capturing this IoT data, which is crucial for subsequent quality assessments. Analysis of the structure of the event log and the derived process is provided through a python notebook[6].

MERODE Domain Model. In [12], we utilized a MERODE Domain Model to develop a Digital Twin for manufacturing applications. Building on this, we have implemented the domain model within Dolly to generate *Prescriptive Models* for implementing Digital Process Twins.

After discovering the actual process, the MERODE Domain Model is instantiated and mapped to the business objects participating. For example, creating instances of the *Device* class allows real-time retrieval of data, status, and actions from physical devices. These digital models, formed by class diagrams, finite state machines and object event tables, are dynamically synchronized with business objects at the business process level, capturing real-time and historical data. Each business object is "tracked" by these models, and every action it performs is updated both in the process and in the domain model instances. This ensures that both real-time and past data produced by business objects can be retrieved for analysis. A representation of the MERODE Domain Model is shown in Figure 4. Further details on the specification and instantiation of the MERODE IoT Domain Model are available online [7].

Data-Driven Process Simulation. At this stage, we assumed the involvement of business experts to introduce changes to the actual process. For this use case, we address the question: *How can we modify the process to reduce total costs and cycle time while maintaining the same operational efficiency?*

Using *bpmn.io*, a BPMN modeler embedded in Dolly, changes were manually applied to the digital process replica (*P1*), resulting in an adapted version, *P1'*. This adaptation aimed to simplify the workflow, reduce costs, and shorten the process cycle time. In *P1'*, the container is moved to the Control Area only if it fails the quality check, eliminating unnecessary movements. Figure 3c depicts the adapted version of the process *P1'*.

The event log was then used to discover the optimal simulation parameters for the process simulation model *P1'* using SimuBridge [23]. SimuBridge integrates components such as control flow, activity duration, and resource utilization by

[6] `https://dub.sh/BPDiscovery-ipynb`
[7] `https://github.com/IvanComp/Dolly/blob/main/README.md`

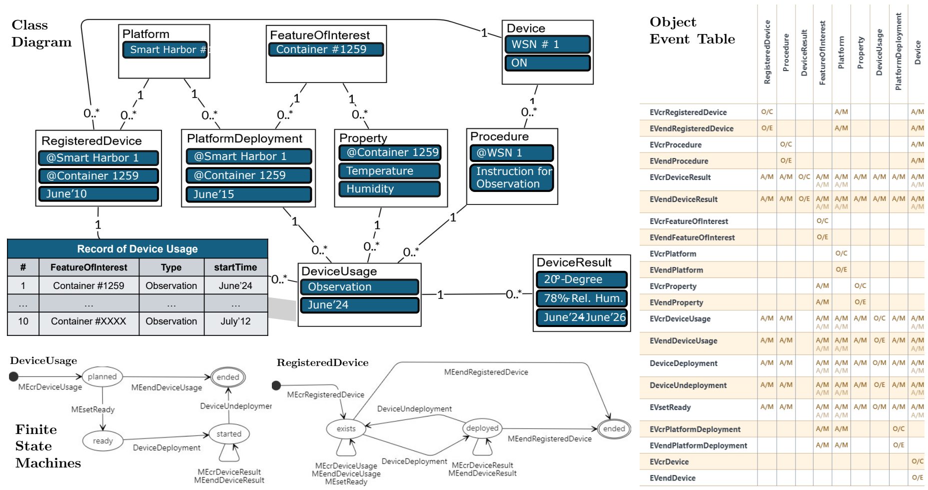


Fig. 4: MERODE Domain Model Instantiation.

analyzing a *.xes* event log file. It leverages the Simod mining algorithm [7], which enhances accuracy to derive models and simulation parameters from event logs. The simulation parameters mined are as follows: Worker 1 handles container system registration and quality checks, earning €20 per hour; Worker 2 manages container movements, earning €30 per hour; Worker 3 performs manual quality inspections, earning €25 per hour. Figure 3b shows these parameters, with each task differentiated by shapes and colors based on the resource associated. For all workers, a 24/7 working timetable was considered. Activities have varying durations: the container quality check takes 1 hour while recording and checking container data in the system takes 5 minutes. To better reflect reality, container movements follow a uniform time distribution, varying between 25 and 35 minutes. Containers have an 84% chance of passing the quality test, as required by the XOR gateway for outgoing sequence flows (84% OK, 16% NOT OK). Additionally, a fixed distribution time is assigned to each instance's arrival.

Finally, we simulated *P1'* using BIMP UI, a scalable and fast BPMN simulator and compared the KPIs derived from *P1* and *P1'*. As motivated by the Digital Process Twin architecture, we employed a hybrid approach integrating mined optimal simulation parameters, real-time data from the MERODE Domain Model, and manual changes to the actual process *P1*. This method continuously updates and reflects the process model, resulting in a data-driven simulation model *P1'*.

Results Evaluation. Table 1 presents the KPIs for *P1* and *P1'*. The KPIs are categorized into cycle time distribution, cost distribution, and resource utilization. In terms of cycle time distribution, *P1'* shows significant improvements with reduced minimum, maximum, and average cycle times, indicating a more efficient process. Regarding cost distribution, *P1'* showed cost savings, with lower minimum, maximum, and average costs, and a significantly reduced total cost,

Legend: ↓ *Reduction*, ↑ *Increase.* — **Hrs:** Hours, **Wks:** Weekends.

KPIs	Original Scenario *(P1)*				Adapted Scenario *(P1')*			
	Min.	Max.	Avg.	Total	Min.	Max.	Avg.	Total
Cycle Time Distr.	3.7 Hrs	15.6 Hrs	11.7 Hrs	12.6 Wks	1.5 Hrs ↓	9.2 Hrs ↓	5 Hrs ↓	12.4 Wks ↓
Cost Distr.	€ 68.20	€ 110.40	€ 80.10	€ 40,054.20	€ 40.40 ↓	€ 93.70 ↓	€ 53.50↓	€ 26,767.50 ↓
	Worker 1	Worker 2	Worker 3	Total	Worker 1	Worker 2	Worker 3	Total
Resource Utiliz.	58.75%	2.00%	3.44%	64.19%	37.54% ↓	1.98% ↓	3.92% ↑	43.44% ↓

Table 1: KPIs of the Simulation for *P1* and *P1'*.

indicating better time efficiency and cost-effectiveness. In resource utilization, *P1'* shows mixed results. Workers 1 and 2 have significantly improved utilization, while Worker 3's workload increases. A graphical comparison between the KPIs values of *P1* and *P1'* is shown in Figure 5 and available online[8]. The *P1* and *P1'* versions of the simulation models, the SimuBridge project file and the simulation results are available online [21][9]. The Dolly source code and instructions for running are available online[10].

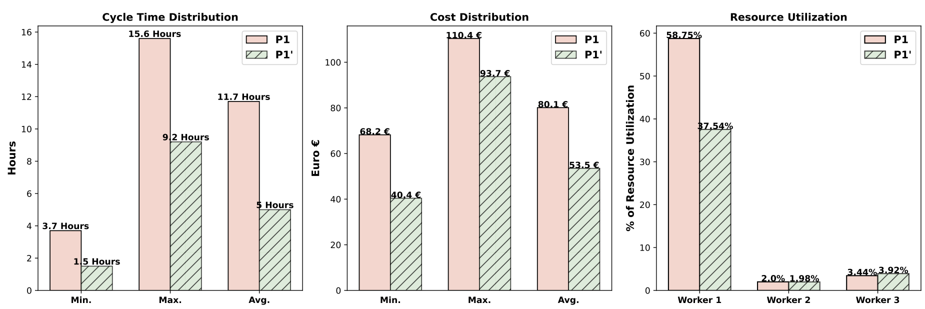


Fig. 5: Comparison of the KPIs for *P1* and *P1'*.

6 Related Work

The concept of Digital Twin has been extensively explored across various domains and purposes [19, 24]. Significant research has focused on implementing Digital Twins in industrial sectors, particularly in replicating and simulating machines and devices used in manufacturing processes [3, 24, 26].

Despite the growing interest, only a limited number of research works focus on implementing Digital Process Twins. For instance, [34] proposes a micro-service

[8] https://bit.ly/Dolly_AnalisysResults_ipynb
[9] https://zenodo.org/records/12671621
[10] https://github.com/IvanComp/Dolly

architecture to integrate physical IoT entities into IoT-Enhanced Business Processes. This approach uses a model-driven development method that combines BPMN models and Digital Twins Definition Language models via Java microservices, allowing IoT virtual replicas to be integrated into real-world processes. However, it lacks capabilities for continuous optimization and adaptation of IoT-Enhanced Business Processes. Similarly, PROWIN [13] is a framework designed for monitoring and executing IoT-Enhanced Business Processes in a multi-robot scenario. It uses the Gazebo Simulator for 3D visualization of the operating scenario and the process's evolution, offering a detailed view of the system's execution. Nonetheless, it does not address the specifics of the software infrastructure needed for maintaining runtime synchronization with the real world. In [5], authors present a framework for managing IoT-Enhanced Business Processes. This solution extends the BPMN standard and integrates models for analysis, featuring a model-to-text transformation engine, an interaction broker for IoT infrastructure, a simulation engine, and a business process engine. However, it lacks detailed real-time communication with physical counterparts.

This work advances the state of the art by proposing a novel conceptual architecture for deploying Digital Process Twins. It outlines a procedure for creating, managing, and simulating digital replicas of business processes to assess potential changes before real-world implementation. Additionally, we introduce Dolly, a prototype framework that allows to implementation of the proposed Digital Process Twin conceptual architecture.

7 Conclusion

In this paper, we moved a first step in introducing a novel conceptual architecture for deploying Digital Process Twins to enhance resilient process changes and support informed decision-making through predictive insights derived from data-driven process simulations. The architecture integrates heterogeneous digital models (e.g., descriptive, predictive, and prescriptive) to design, synchronize, and simulate a high-fidelity digital replica of business processes, leveraging data extracted from real-time business object monitoring and process mining analysis.

The conceptual architecture promotes a feedback loop mechanism that utilizes data-driven process simulation on the process replica to continuously assess the potential impacts of desired process changes. If process performance improves or remains stable, these changes are considered to be actuated by business experts in the real-world process. Moreover, conducting tests on digital replicas enables secure, risk-free evaluation of changes, thereby reducing deployment costs and accelerating process updates. The approach was evaluated using Dolly, an early-prototype framework implementing the proposed Digital Process Twins architecture in the context of a container dispatching process, revealing significant improvements.

In future research, we aim to improve further Dolly, currently in its prototype stage, focusing on performance in high data volume environments, such as typical IoT settings. We also plan to test Dolly in more complex and larger-scale scenarios for more accurate and realistic evaluation.

References

1. van der Aalst, W.M.P.: Business Process Simulation Survival Guide. In: BPM, Introduction, Methods, and Information Systems, pp. 337–370. Handbooks on Information Systems, Springer (2015)
2. van der Aalst, W.M.P., Carmona, J. (eds.): Process Mining Handbook, Lecture Notes in Business Information Processing, vol. 448. Springer (2022)
3. Becker, F., et al.: A Conceptual Model for Digital Shadows in Industry and Its Application. In: Conceptual Modeling. pp. 271–281. Springer International Publishing (2021)
4. Beerepoot, I., et al.: The biggest business process management problems to solve before we die. Computers Industry **146**, 103837 (2023)
5. Bocciarelli, P., D'Ambrogio, A., Panetti, T.: A Model Based Framework for IoT-Aware Business Process Management. Future Internet **15**(2) (2023)
6. Camargo, M., Dumas, M., González-Rojas, O.: Automated discovery of business process simulation models from event logs. Dec. Sup. Syst. **134**, 113284 (2020)
7. Camargo, M., Dumas, M., Rojas, O.G.: Simod: A tool for automated discovery of business process simulation models. In: BPM Demo. CEUR Workshop Proceedings, vol. 2420, pp. 139–143. CEUR-WS.org (2019)
8. Compagnucci, I., Corradini, F., Fornari, F., Polini, A., Re, B., Tiezzi, F.: Modelling Notations for IoT-Aware Business Processes: A Systematic Literature Review. In: BPM Workshops. LNBIP, vol. 397, pp. 108–121. Springer (2020)
9. Compagnucci, I., Corradini, F., Fornari, F., Polini, A., Re, B., Tiezzi, F.: A systematic literature review on IoT-Aware Business Process Modeling Views, Requirements and Notations. Software and Systems Modeling **14**(1), 1–36 (2022)
10. Compagnucci, I., Corradini, F., Fornari, F., Re, B.: Trends on the Usage of BPMN 2.0 from Publicly Available Repositories. In: Perspectives in Business Informatics Research. LNBIP, vol. 430, pp. 84–99. Springer (2021)
11. Compagnucci, I., Corradini, F., Fornari, F., Re, B.: A Study on the Usage of the BPMN Notation for Designing Process Collaboration, Choreography, and Conversation Models. Business & Information Systems Engineering **66**, 43–66 (2024)
12. Compagnucci, I., Snoeck, M., Asensio, E.S.: Supporting Digital Twins Systems Integrating the MERODE Approach. In: 2023 ACM/IEEE International Conference on Model Driven Engineering Languages and Systems Companion (MODELS-C). pp. 449–458 (2023)
13. Corradini, F., Pettinari, S., Re, B., Rossi, L., Tiezzi, F.: Executable Digital Process Twins: Towards the Enhancement of Process-Driven Systems. Big Data and Cognitive Computing **7**(3) (2023)
14. Depaire, B., Martin, N.: Data-driven process simulation. In: Sakr, S., Zomaya, A.Y. (eds.) Encyclopedia of Big Data Technologies. Springer (2019)
15. Dumas, M.: Constructing Digital Twins for Accurate and Reliable What-If Business Process Analysis. In: Conference on Business Process Management. CEUR Workshop Proceedings, vol. 2938, pp. 23–27 (2021)
16. Dumas, M., La Rosa, M., Mendling, J., Reijers, H.A.: Fundamentals of Business Process Management. Springer (2018)

17. Eramo, R., Bordeleau, F., Combemale, B., van den Brand, M., Wimmer, M., Wortmann, A.: Conceptualizing digital twins. IEEE Softw. **39**(2), 39–46 (2022)
18. Fornari, F., et al.: Digital Twins of Business Processes: A Research Manifesto. Internet of Things **30**, 101477 (2025)
19. Grieves, M.W.: Digital Twins: Past, Present, and Future, pp. 97–121. Springer International Publishing (2023)
20. Grimm, J., Kraus, A., van der Aa, H.: CDLG: A tool for the generation of event logs with concept drifts. In: BPM Demo. CEUR Workshop Proceedings, vol. 3216, pp. 92–96. CEUR-WS.org (2022)
21. Ivan Compagnucci, Barbara Re, Monique Snoeck, and Estefanía Serral: Open Science Artifact: Performance Analysis of Architectural Patterns for Federated Learning Systems. `https://zenodo.org/records/12671621` (2024)
22. Kerremans, M., Sugden, D., Duffy, N.: Magic quadrant for process mining platforms. `https://www.gartner.com/doc/reprints?id=1-2HEH7GJM&ct=240426&st=sb` (2024), Gartner, Inc. - Last Acces 8 July
23. Leon, B., Klessascheck, F., Nepeina, S., Warmuth, C., Kampik, T., Pufahl, L.: Simubridge: Discovery and management of process simulation scenarios. In: BPM Demo. CEUR Workshop Proceedings, vol. 3469, pp. 77–81. CEUR-WS.org (2023)
24. Liu, M., Fang, S., Dong, H., Xu, C.: Review of digital twin about concepts, technologies, and industrial applications. Manufacturing Systems **58**, 346–361 (2021)
25. Rabe, M., Kilic, E.: Framing the digital business process twin: From a holistic maturity model to a specific and substantial use case in the automotive industry. In: BPM Workshops. pp. 353–364. Springer Nature Switzerland (2024)
26. Rasheed, A., San, O., Kvamsdal, T.: Digital twin: Values, challenges and enablers from a modeling perspective. IEEE Access **8**(1), 21980–22012 (2020)
27. Reichert, M., Weber, B.: Enabling Flexibility in Process-Aware Information Systems - Challenges, Methods, Technologies. Springer (2012)
28. Rosenthal, K., Ternes, B., Strecker, S.: Business process simulation on procedural graphical process models. Bus. Inf. Syst. Eng. **63**(5), 569–602 (2021)
29. Saragiotis, P.: Business Process Management in the port sector: a literature review. Maritime Business Review **4** (2019)
30. Snoeck, M., Dedene, G.: Existence dependency: The key to semantic integrity between structural and behavioral aspects of object types. IEEE Transactions on Software Engineering **24**(4), 233–251 (1998)
31. Snoeck, M.: Enterprise Information Systems Engineering - The MERODE Approach. The Enterprise Engineering Series, Springer (2014)
32. Snoeck, M., Verbruggen, C., De Smedt, J., De Weerdt, J.: Supporting Data-Aware processes with MERODE. Software and Systems Modeling (2023)
33. Torres, V., Serral, E., Valderas, P., Pelechano, V., Grefen, P.: Modeling of IoT devices in Business Processes: A Systematic Mapping Study. In: Business Informatics. pp. 221–230. IEEE (2020)
34. Valderas, P.: Supporting the Implementation of Digital Twins for IoT-Enhanced BPs. In: Information Science and the Connected World. LNBIP, vol. 476, pp. 222–238. Springer (2023)
35. Vemuri, P., Poelmans, S., Compagnucci, I., Snoeck, M.: Using Formative Assessment and Feedback to Train Novice Modelers in Business Process Modeling. In: ACM/IEEE International Conference on Model Driven Engineering Languages and Systems Companion (MODELS-C). pp. 130–137 (2023)