Process Mining for e-Government collaborative processes choreography: a case study

Lucía Antunes[1], Andrea Delgado[1], and Laura González[1]

Instituto de Computación, Facultad de Ingeniería,Universidad de la República
{lucia.antunes,adelgado,lauragon}@fing.edu.uy

Abstract. Digital Government processes are distributed between several public organizations that work together to provide services to citizens. Integration of both internal orchestration-like processes and heterogeneous technologies is a key challenge for such collaborative processes. The Uruguayan Digital Government Agency (AGESIC) provides, among other components, a centralized Interoperability Platform (PDI), which is an environment that facilitates integration and connectivity between participant organizations. Messages exchanged between collaborative processes participants travel through the PDI, thus registering process choreography data. In this paper, we present a case study we carried out using messages data registered in the PDI to discover BPMN 2.0 choreography models for selected processes. We adapted an existing correlation algorithm and generated a collaborative event log that includes participants with corresponding messages flow as attributes, from which to obtain the choreography model. These choreography models have proven to be useful for the business area in analyzing the real interaction between public organizations against the expected one, detecting deviations from the intended use of the platform.

Keywords: Digital Government processes · collaborative processes · collaborative process choreography.

1 Introduction

Business processes (BPs) within organizations define the activities and their sequence within an organizational and technological environment, that provides expected results or outcomes to fulfill a business objective [16]. Traditional intra-organizational BPs are also referred as orchestration-like processes, where the control flow is in charge of the owner organization and within its limits. On the other hand, collaborative inter-organizational BPs have gained increasingly interest in the last decade, posing several challenges over orchestration-like processes in all phases of the BPs lifecycle. Collaborative BPs involve more than one organization that interacts with each other via message exchanges, to obtain a global and coordinated result of interest. This type of processes, can have different forms and distribution options between participant organizations [1].

Digital Government processes are distributed between several public organizations that work together to provide services to citizens. Integration of both internal orchestration-like processes and heterogeneous technologies is a key challenge for such collaborative processes. The Uruguayan Digital Government Agency (AGESIC)[1] provides a centralized Interoperability Platform (PDI), which is an environment that facilitates integration and connectivity between participant organizations. Messages exchanged between collaborative processes participants travel through the PDI, enabling the registration of process choreography metadata. Also, a traceability system component is used by participant organizations to centrally register the activities they carry out within BPs execution.

In previous work [5] we have analyzed orchestration-like and collaborative processes from the traceability system, using real data provided by AGESIC to get insights from their processes. If a BP contains an invocation from one participant organization to another, it is expected that activities are registered within the traceability system, and the message exchanges are carried out through the PDI. In this context, the primary goal of this paper is to explore the feasibility of applying process mining techniques [2] to discover collaborative process choreographies from the PDI message logs, which lack explicit case identifiers. We aim to reconstruct event traces that accurately reflect real-world inter-organizational procedures and to evaluate whether the discovered models are useful for analyzing conformance and detecting deviations.

To this end, we present a case study we carried out using messages metadata registered in the PDI to discover BPMN 2.0 choreography models for selected processes. We adapted an existing correlation algorithm [11] to handle the specific challenges of the PDI data and generated a collaborative event log for choreographies [7] that includes participants with corresponding messages flow as attributes, from which to obtain the choreography process model, using an approach previously developed [13]. We evaluate the validity of the results by comparing the discovered message sequences to known procedures and through statistical validation where possible.

The rest of this paper is structured as follows: in Section 2 we present concepts on collaborative processes and choreografies as basis for the paper and in Section 3 we discuss related work. In Section 4 we present the adaptation of an existing correlation algorithm we selected. Then, in Section 5 we present the case study we carried out with the real data of selected processes from the PDI. Finally, in Section 6, we provide conclusions and discuss future work.

2 Background

This section provides background on collaborative processes, BPMN choreography diagram, and on the PDI.

2.1 Collaborative Processes and BPMN Choreography Diagram

Business processes performed by one organization may interact with processes performed by other organizations, leading to collaborative business processes.

[1] `https://www.gub.uy/agencia-gobierno-electronico-sociedad-informacion-conocimiento/`

The interaction between business processes in different organizations is achieved by message exchanges and it is specified in a process choreography [16].

Figure 1 presents a collaborative business process specified with a BPMN choreography diagram. This type of diagram focuses on message exchanges between participants. The initiating participant of a choreography activity (e.g. Place Order) is the one that is not shaded in grey in the diagram.

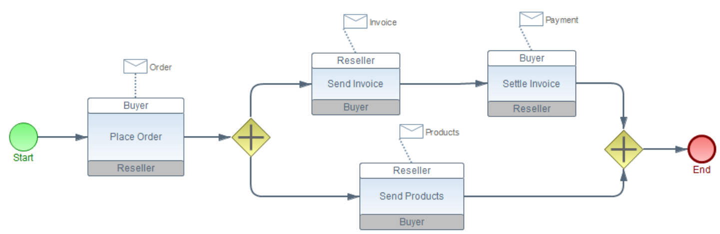

Fig. 1. Collaborative Business Process - BPMN Choreography Diagram

2.2 Interoperability Platform (PDI)

The Interoperability Platform (PDI) [8], provided by AGESIC, enables government organizations to offer business services by leveraging web services. These services, typically hosted on the organizations' own infrastructure, are exposed and accessed through proxy services deployed on the PDI. As a result, the platform can process all service invocations and record related metadata.

In order to illustrate this operation, Figure 2 presents how web services messages are processed by the PDI when the Ministry of Public Health (MSP) invokes, via a proxy service, the Basic Information Service provided by the Civil Identification National Directorate (DNIC). This service receives a citizen identifier and returns basic information about citizens (e.g. names).

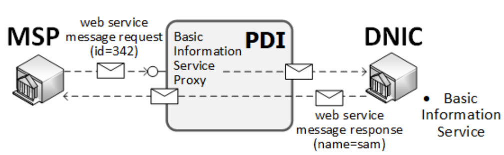

Fig. 2. Interoperability Platform (PDI).

In this context, two or more government organizations may carry out collaborative business processes by leveraging the services available in the platform.

3 Related work

This section presents a review and analysis of existing methods and related work on event correlation in execution logs. Most researchs assume that events can be grouped into process instances using a case identifier, while methods that do not require this are less common. Based on this analysis, three representative techniques were selected for their approach to the characteristics of the environment and data considered in this work.

One class of approaches explores the use of machine learning techniques to infer trace boundaries based on event sequences. A representative example is the method proposed in [12], which uses Word2Vec embeddings to model relationships between events. Specifically, the CBOW (Continuous Bag of Words) technique is applied to sequences of user interactions or system transitions, assuming that events from the same instance occur in temporal proximity. The method involves generating artificial training data from a known transition graph (e.g., screens or actions in a system), labeling boundaries between traces, and training a neural model to predict potential cut points in unseen sequences. Once trained, the model estimates the probability that an event is the start of a new instance, allowing the segmentation of raw logs into process traces. This approach is particularly useful in user-centric systems where interactions are temporally contiguous and follow recognizable patterns.

The proposal in [15] focuses on building process models from raw, unsegmented logs and then using these models to guide trace reconstruction. It operates in two main phases. First, it constructs an acyclic process model by applying integer linear programming to a graph of observed events, using precedence and duration matrix to capture structural and temporal patterns. The model is then used in a second phase, where trace reconstruction is formulated as a quadratic optimization problem. Candidate pairs of events are evaluated for consistency with the model, and binary decision variables are introduced to represent possible successions.

The algorithm in [11] is specifically designed to correlate events in logs of web systems interactions. It is structured into four main stages: i) Identification of Relevant Attributes: Event logs are examined to extract key attributes, possible identifiers, timestamps with close temporal proximity, and message structures that may serve as indicators for correlating events; ii) Generation of Correlation Rules: Correlation rules are defined using different types of conditions: atomic, conjunctive, and disjunctive. Atomic conditions verify whether a single attribute matches between two events, such as an ID or timestamp. Conjunctive conditions consist of multiple atomic conditions that must all be true simultaneously for a correlation to be established. Disjunctive conditions allow correlation if at least one condition is met, providing greater flexibility for detecting broader and less restrictive correlations; iii) Rule Optimization: Redundant or insignificant rules are filtered out, and heuristics are applied to enhance the precision of the correlation results; and iv) Evaluation and Adjustment: The traces generated by the correlation rules are validated by comparing them against the expected process structures.

Regarding the discovery of choreographies in BPMN 2.0, to the best of our knowledge [13] was the first proposal, then extended in [14]. The approach uses a collaboration event log extended for the choreography perspective, including only messages exchanged and participants from and to as sender and receivers of such messages to discover the control flow of the choreography in an orchestration-like way. Similarly [10] propose identifying the correlation between events at the case and activity levels, using a set of rules to identify and mark messages exchanged, extending [9] which was originally limited to two participants and the discovery of the collaboration only, adding choreographies discovery in BPMN 2.0.

4 Correlation Algorithm Based on Attributes and Rules

The implemented correlation algorithm is based on the algorithm in [11], which is designed to correlate events originating from web systems in contexts where explicit case identifiers are not available. Based on the analysis of the three selected algorithms presented in Section 3, this algorithm was selected primarily because it is specifically designed to handle interactions in web service environments. Additionally, its flexible, rules-based approach enables its adaptation to the structure and characteristics of the input records used in this study.

This problem is particularly relevant in environments such as e-Government platforms where multiple organizations interact through web services without centralized identifiers linking their communications. Reconstructing traces in such contexts enables better understanding, monitoring, and improvement of collaborative processes. **Our proposal comprises** a practical adaptation and application of [10] to the specific context of e-Government collaborative processes within the PDI. This adaptation enables the reconstruction of coherent event traces from raw message logs, producing collaborative event logs suitable for BPMN 2.0 choreography discovery.

4.1 Working Context

The algorithm operates over logs containing raw event records collected from the PDI involving multiple organizations. The logs do not contain explicit case identifiers and present high heterogeneity, significant volume, and possible noise.

The dataset used corresponds to logs provided by AGESIC covering the years 2020 to 2022, with an approximate volume of 1,7 TB. This large-scale data highlights the complexity and challenges involved in processing and correlating events across multiple organizations, in distributed collaborative processes, making it necessary to work progressively with smaller date ranges to ensure manageable analysis and efficient processing. For reference, processing one month of log data is just under 15 minutes.

4.2 Correlation Algorithm Design and Implementation

The adapted method follows three main stages, each defined to reconstruct event sequences based on selected attributes and correlation rules, consolidating the four original stages to streamline the process and ensuring that the correlation process is aligned with the specific data patterns and attributes of our case study.

1. **Exploratory analysis:** Focused on the relevance and utility of identified attributes for correlating events, such as partial identifiers, value matching, time proximity, and message patterns.
2. **Definition of Correlation Rules:** Generation of rules combining atomic conditions (evaluating individual attributes to filter out noise or irrelevant events), conjunctive conditions (simultaneous constraints on sequences such as temporal restrictions, minimum length, and frequency), and disjunctive conditions (semantic criteria requiring at least one event in a sequence to satisfy certain attribute values). These rules are optimized using heuristics based on frequency and distribution to enhance discriminative power.
3. **Evaluation and Adjustment:** Validation of the algorithm's effectiveness through a case study using real event logs.

The proposed solution combines temporal segmentation with a model based on rules applied to event attributes. A minimum repetition frequency is also incorporated to filter out irrelevant or infrequent patterns. The goal is to reconstruct coherent event sequences that reflect the execution of distributed collaborative processes based on the messages exchanged.

Exploratory Analysis and Data Preparation. In the first stage, an exploratory analysis was performed to transform raw records into a structured and human-readable representation. Python was initially used due to its flexibility in parsing large volumes of data efficiently. Parsed data were stored in a PostgreSQL database for initial exploration and visualization of a reduced dataset spanning one month. Later, for processing a larger dataset, Python was again employed directly on files, offering better performance for this scale.

This analysis allowed identification and extraction of relevant event attributes such as timestamps, partial identifiers, requested services, requesting organizations, and IP addresses, as well as analysis of their relationships. Traditional identifiers like TRANSID and GTID—which correspond to the transaction and global transaction identifiers within the PDI infrastructure—were observed to be insufficient to reconstruct complete traces across distributed systems. This observation motivated the selection of additional attributes with higher potential to establish significant correlations in collaborative, multi-organizational processes.

Formal Adaptation of Correlation Rules. In this implementation, correlation rules are adapted focusing on the data semantics and declared as follows:

– **Atomic Rules:** Filter individual events considered noise or irrelevant based on the presence of specific values in certain attributes. Formally:

$$r_a(e) = \begin{cases} 0, & \text{if } v(e, a_i) \in V_{\text{exclude}} \\ 1, & \text{otherwise} \end{cases}$$

where $v(e, a_i)$ is the value of attribute a_i in event e, and V_{exclude} is the set of values indicating exclusion.

- **Conjunctive Rules:** Establish simultaneous conditions that a sequence s must satisfy, such as temporal constraints, minimum length, and minimum frequency:

$$r_c(s) = r_1(s) \wedge r_2(s) \wedge \cdots \wedge r_n(s)$$

where each $r_i(s)$ is a condition over the sequence.

- **Disjunctive Rules:** Define criteria in which at least one event in a sequence must satisfy certain attribute values to identify semantically relevant characteristics for correlation:

$$r_d(s) = \bigvee_{e \in s} \bigvee_{i} (v(e, a_i) = v_{\text{ref},i})$$

where $v_{\text{ref},i}$ is the expected reference value for attribute a_i.

This adaptation aims to align algorithm rule formalization with the nature of the data and the objectives of this case of study.

Implementation and Processing. The algorithm processes candidate sequences generated through a temporal window that groups consecutive events with timestamps differing less than a predefined threshold (e.g. 5 minutes). These temporal constraints, along with minimum sequence count, minimum frequency lenght, exclusion and inclusion filters according to the previously defined correlation rules—are integrated within the atomic, conjunctive, and disjunctive correlation rules. Attribute-based correlation is applied at this stage by enforcing the previously defined atomic, conjunctive, and disjunctive rules on the event attributes. Specifically, atomic rules are applied during the preprocessing step (parse_and_filter) via the exclusion_filter, conjunctive rules are enforced during the main iteration over events, and disjunctive rules are applied through the inclusion_filter, requiring that at least one event in a sequence satisfies semantic attribute conditions. The temporal window threshold is adapted according to the characteristics of the process under analysis, ensuring that event sequences reflect the execution timing as realistically as possible. Only sequences satisfying all these combined rule criteria are considered.

Recurrent sequences identified are stored for subsequent analysis, producing a set of candidate traces representing collaborative behavior among systems. The pseudocode shown in Listing 1 summarizes the general logic of the algorithm reflecting the ordered steps.

Trace Generation and Export in XES Format. As a final step, a function was developed to extract occurrences of the detected representative sequences and export them as files in XES format. This format, widely used in process mining, facilitates interoperability with various specialized tools such as ProM, which was employed in this work for model validation. Each generated trace consists of a sequence of events including standard XES schema attributes-such as activity name, timestamp and participating resource- as well as customized attributes like source and destination participant which are specifically required by the ProM choreography discovery plugin. This ensures that traces are fully compatible for accurate reconstruction of the collaborative process models.

```
Input:
  log, time_limit, min_length,
  min_frequence, inclusion_filter, exclusion_filter

Preprocessing:
  events = parse_and_filter(log, exclusion_filter, inclusion_filter)
  sort(events)

Initialize:
  candidate_sequences = []
  recurrent_sequences = []
  processed = set()

For each current_event in ordered_events:
  If current_event in processed:
    continue

  sequence = [current_event]
  processed.add(current_event)
  previous_time = current_event.timestamp

  For each next_event after current_event:
    If next_event not in sequence and next_event not in processed:
      If (next_event.timestamp - previous_time) <= time_limit:
        sequence.append(next_event)
        processed.add(next_event)
        previous_time = next_event.timestamp

  If length(sequence) >= min_length:
    candidate_sequences.append(sequence)

For each candidate_sequence:
  If candidate_sequence.frequency >= min_frequence:
    recurrent_sequences.append(candidate_sequence)

Output:
  recurrent_sequences
```

Listing 1. Event Correlation Algorithm Pseudocode

5 Case Study

To effectively apply, verify the correlation algorithm and detect relevant sequences within the log containing a significant volume of records, the definition of adequate filters is essential. Since events from different procedures may be interleaved within the same time interval, attribute-based correlation and appropriate algorithm parametrization are particularly important to separate and extract valid, coherent traces.

The filter definition is based on a dual approach. On one hand, frequent sequences emerging after preliminary algorithm execution are considered, with special attention to repetitions and patterns potentially indicating structured processes. On the other hand, a grouping and statistical counting of key attribute values present in events is performed without associating them to particular instances, allowing identification of numerical relationships and potential matches between pairs of keys. This combination of statistical information and partial results guides the selection of inclusion filters for the algorithm.

From this strategy, candidate sequences are obtained and attempts are made to map them to known real procedures. Although communication is asynchronous as in real-world distributed systems; as the time gap between sending and receiving is extremely small —often just a few milliseconds— while the logs record time at seconds, both request and response are treated as a single message (part of the same event) to simplify the analysis without losing relevant information.

A significative case is the sequence detected associated with the Unique State Suppliers Registry (RUPE), involving multiple governmental entities such as the Ministry of Industry, Energy and Mining (MIEM), the General Tax Directorate (DGI), the State Insurance Bank (BSE), the e-Government Agency (AGESIC), and the State Procurement Regulatory Agency (ACCE). The participation of diverse organizations in this sequence enriches the analysis by enabling the study of a trace that reflects communication and multiple exchanged messages.

This process corresponds to the steps that ACCE performs to verify that a potential state provider accomplishes all necessary requirements to be officially authorized as such. Once the candidate sequence was extracted, it was parsed into the XES event log format extended for choreographies [7] to enable process mining analysis. The resulting choreography model was obtained using a ProM plug-in [3] developed in previous work [13,4], and is depicted in Figure 3. It shows that the exchanged messages are executed in parallel, as the control of required certificates does not follow a strict sequential order. Table 1 summarizes the sequence of messages sent by ACCE from Web services request to destinations.

Table 1. RUPE Web Service requests

Provider	**Web Service**	**Operation**
AGESIC	ProviderUpdates	ProviderUpdate
DGI	CVA_WS	ExecuteCVA
BPS	GetCertificate	GetCommonCertificate
BSE	ADTCertificate	dayCertificate
MIEM	PYME_Certificate	Get

Other relevant cases were analyzed to evaluate the correlation algorithm, one such case was the Passport Application, where two messages were identified as part of the process:

- **MI** → **DNIC**: Return personal data based on the identity identifier number.
- **DGREC** → **DNPT**: Request to verify if the person has judicial records.

This aligns with the procedure's logic, which first retrieves complete personal data using the identity identifier, and then communicates with the technical police to obtain the judicial records certificate to proceed with the process.

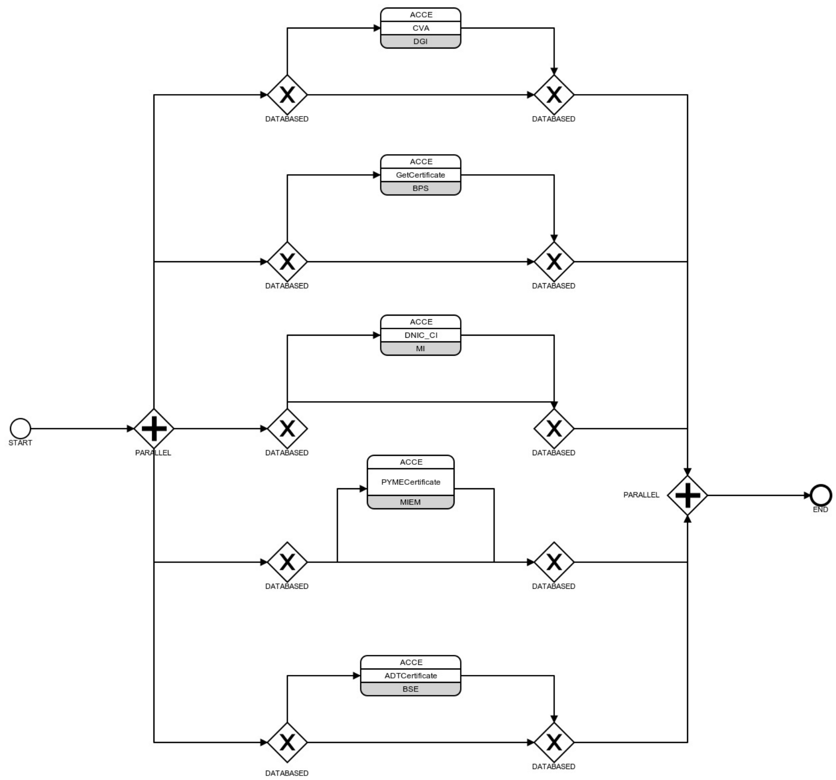

Fig. 3. Choreography of RUPE Requests

Another studied process was the Birth Certificate (CNV). Based on previous research [6], information about message exchanges in this process was available. This process results interesting for containing sequential messages involving multiple participants, such as the Ministry of Public Health (MSP), AGESIC, and General Directorate of Civil Registry (DGREC). Although messages highly correlated with the procedure and its receiving organizations were found, the complete trace was not detected. This could be due to the procedure covering a longer time window than the one tested. Extending the time window was challenging given the high volume of data.

The most significant exchanged message formalizing the procedure is the "CNV Certificate" between MSP and DGREC, which allowed statistical validation by comparing the birth counts in Uruguay with the number of occurrences of this record, showing similar results. This validation could confirm that this message is indeed the correct one representing the procedure. Other records could not be numerically validated but are generic and belong to multiple processes, showing a higher number of occurrences. The following are some of the key message exchanges identified in this process:

- **DGREC** → **MSP**: Registration of the newborn.
- **DGREC** → **AGESIC**: Notification of people updates.
- **DNIC** → **AGESIC**: Request notification of people updates.

Finally, the Equity Plan procedure was investigated, a monthly monetary benefit granted by MIDES and BPS to vulnerable households. Two messages exchanged between these institutions were identified, related to verifying the vulnerability status of the beneficiary and executing the corresponding payment service:

- **BPS** → **MIDES**: Indicate if a person is socioeconomically vulnerable.
- **MIDES** → **BPS**: Execution of the web service related to the equity plan.

Although the presented case study focuses on a limited set of processes, the proposed correlation approach demonstrates high scalability potential. With minor adjustments to the rules and, in particular, with additional process information incorporated into the conjunctive and disjunctive rules, the algorithm could reconstruct multiple traces simultaneously.

6 Conclusions

This work presented an exploratory study on the feasibility of detecting and correlating traces of collaborative BPs with real data from the PDI platform of the e-Government that registers messages interaction between public organizations. We adapted an existing correlation algorithm specifically designed to handle interactions in web service environments, to the context of our data, combining temporal segmentation with a model based on rules applied to event attributes, filtering out irrelevant or infrequent patterns.

The analysis carried out showed that the metadata registered for each message, while reduced, could be used as basis for the identification of choreographies and corresponding process model discovery. However, choreographies detected were not always completed due to missing data or issues with data registration, that will be useful to improve interactions registration within the PDI.

We acknowledge existing limitations of our work. In the first place it is a case study single-site, and although information on the PDI and web services exposed and how to use them is public information published in the website of the agency, we cannot provide the event logs as-is due to privacy issues. Also, although we evaluated several correlation algorithms we selected one and adapted it to our needs, with no comparison with other options. As future work we would like to evaluate whether other correlation algorithms could be adapted to our data, and compare results with the ones already obtained.

Acknowledgments

Supported by project "Minería de procesos y datos para la mejora de procesos colaborativos aplicada a e-Government" funded by Agencia Nacional de Investigación e Innovación (ANII), Fondo María Viñas (FMV) "Proyecto ANII N° FMV_1_2021_1_167483", Uruguay.

References

1. van der Aalst, W.M.P.: Intra- and inter-organizational process mining: Discovering processes within and between organizations. In: The Practice of Enterprise Modeling - 4th IFIP WG 8.1 Working Conference, PoEM 2011. LNBIP, vol. 92, pp. 1–11. Springer, Cham (2011). https://doi.org/10.1007/978-3-642-24849-8_1
2. van der Aalst, W.M.P.: Process Mining - Data Science in Action, Second Edition. Springer, Cham (2016). https://doi.org/10.1007/978-3-662-49851-4
3. van der Aalst, W.M.P., van Dongen, B.F., Günther, C.W., Rozinat, A., Verbeek, E., Weijters, T.: Prom: The process mining toolkit. In: Procs of the BPM Demonstration Track. CEUR Workshop Proceedings, vol. 489. CEUR-WS.org (2009)
4. Andrade D., Delgado A., Calegari D.: BPMN 2.0 choreo.process discovery ProM plugin. https://gitlab.fing.edu.uy/open-coal/bpmnchoreographypm (2023)
5. Delgado, A., Calegari: Discovery and analysis of e-government business processes with process mining: a case study. In: Proceedings of the 55 Hawaii International Conference on System Sciences (HICSS). pp. – (2022)
6. Delgado, A., Calegari, D., González, L., Montarnal, A., Benaben, F.: Towards a metamodel supporting e-government collaborative business processes management within a service-based interoperability platform. In: Proceedings of the 53rd Hawaii International Conference on System Sciences (HICSS). p. 2019–2028 (2020)
7. González, L., Delgado, A.: Compliance requirements model for collaborative business process and evaluation with process mining. In: XLVII Latin American Computing Conference (CLEI). IEEE (2021)
8. González, L., Ruggia, R., Abin, J., Llambías, G., Sosa, R., Rienzi, B., Bello, D., Álvarez, F.: A service-oriented integration platform to support a joined-up e-government approach: The uruguayan experience. In: EGOVIS/EDEM 2012. Springer (2012). https://doi.org/10.1007/978-3-642-32701-8_13
9. Hernandez-Resendiz, J.D., Tello-Leal, E., Marin-Castro, H.M., Ramirez-Alcocer, U.M., Mata-Torres, J.A.: Merging Event Logs for Inter-organizational Process Mining, pp. 3–26. Springer (2021). https://doi.org/10.1007/978-3-030-71115-3_1
10. Hernandez-Resendiz, J.D., Tello-Leal, E., Sepúlveda, M.: A data-driven approach to discovering process choreography. Algorithms **17**(5) (2024). https://doi.org/10.3390/a17050188
11. Motahari-Nezhad, H., Saint-Paul, R., Casati, F.e.a.: Event correlation for process discovery from web service interaction logs. The VLDB Journal **20**, 417–444 (2011). https://doi.org/https://doi.org/10.1007/s00778-010-0203-9
12. Pegoraro, M., Uysal, M.S., Hülsmann, T.H., van der Aalst, W.M.P.: Resolving uncertain case identifiers in interaction logs: A user study. In: International Conference on Business Process Management. Springer (2021)
13. Peña, L., Andrade, D., Delgado, A., Calegari, D.: An approach for discovering inter-organizational collaborative business processes in BPMN 2.0. In: Process Mining Workshops. pp. 487–498. Springer (2024). https://doi.org/10.1007/978-3-031-56107-8_37
14. Peña, L., Andrade, D., Delgado, A., Calegari, D.: Inter-organizational collaborative bpmn 2.0 business process discovery. Journal of Intelligent Information Systems (2024). https://doi.org/10.1007/s10844-024-00876-7
15. Pourmirza, S., Dijkman, R., Grefen, P.: Correlation miner: Mining business process models and event correlations without case identifiers. Inf. Systems **62** (2016)
16. Weske, M.: Business Process Management - Concepts, Languages, Architectures, 3rd Edition. Springer (2019). https://doi.org/10.1007/978-3-662-59432-2