CEIR REPORT

No. 02/2024

Social Process Mining: Deriving Collaborative Work Processes from the Event Data of Enterprise Collaboration Systems

CEIR | Center for Enterprise Information Research

process science

SPM-1
Social Process Mining

Center for Enterprise Information Research (CEIR)
University of Koblenz
Universitätsstraße 1
56070 Koblenz
Germany
ceir.de

Research Group Process Science
University of Koblenz
Universitätsstraße 1
56070 Koblenz
Germany
uni-ko.de/ps

CEIR is represented by:
Prof. Dr. Susan P. Williams (williams@uni-koblenz.de)
Prof. Dr. Petra Schubert (schubert@uni-koblenz.de)

Research Group Process Science is represented by:
Prof. Dr. Patrick Delfmann (delfmann@uni-koblenz.de)

CEIR Report No. 02/2024
Title: Social Process Mining: Deriving Collaborative Work Processes from the Event Data of Enterprise Collaboration Systems
Research was conducted by:
Petra Schubert (CEIR)
Jonas Blatt (Research Group Process Science)
Martin Just (CEIR)
Patrick Delfmann (Research Group Process Science)

Publication date: 20/11/2024
ISSN: 2751-5109
URN: nbn:de:hbz:kob7-25276

Acknowledgements:
This work report is part of the Social Process Mining research project. The research project is funded by the *Deutsche Forschungsgemeinschaft* (DFG) (Project No. 445182359).

SUMMARY

The literature contains very few publications on the application of Process Mining methods for the analysis of event logs in Enterprise Collaboration Systems (ECS). This is not surprising because the analysis of digital support for collaborative work is extremely intricate due to various challenges relating to a lack of data access, poor data quality, unstructured processes and a lack of descriptive models. This article reports on the findings from an Action Design Research (ADR) project. The ADR team had access to a large instance of an operational ECS with more than 3000 users. The event log contains several million entries. Together with the platform's operating team, intensive research was carried out over a period of six years on ways of analysing user activities on the platform. Several cycles were run to develop new methods and computational techniques to decipher the event logs and meaningfully describe the processes recorded in them. Thanks to the close collaboration between the researchers and the operators of the collaboration platform, it was possible to compare the real-world processes carried out in the platform with the processes discovered using a novel method for Social Process Mining (SPM). The result is a pattern analysis that discovers patterns in processes that have a high degree of correspondence with the real-world scenes of collaborative work. The research work has now reached a point where other software products are included (multi-system analysis) and a catalogue of collaborative work situations (scenes) has been developed to describe the process patterns that result from the Process Mining and graph-based analysis techniques.

The results of the study show that:

- **Event data from collaboration software requires intricate transformation and enhancement** before being usable in analytics.
- **Log formats of commonly used software products vary significantly**, necessitating a specialized log pre-processor.
- **Application of existing Process Mining techniques depends on a viable CaseID**, with Social Document ID (SocDocID) emerging as a suitable candidate.
- **Process instances of collaboration activities often appear unique**, requiring condensation into recognizable patterns.
- **Semi-automated identification of collaboration patterns** with descriptive labels (stored in an extensible SPM Catalogue) **enables analysis of collaborative work**.
- **Future research needs to perform multi-system analyses and comparisons** of collaborative practices across organisational units.

TABLE OF CONTENTS

TABLE OF FIGURES

1 Introduction

The analysis of digital traces that users leave behind when using business application systems is a topic of increasing interest for both, academics and practitioners (Berente et al., 2019). Today's work in organisations is to a large extent supported by digital technologies. For many years, *structured core business processes* that are geared at the *production* of products and/or services and the supporting activities of *planning and administration* have been the primary target of digital support initiatives (Baptista et al., 2020). As a response to the existing demand on the side of user organisations, there is a mature and saturated market for commercial standard software for *process-aware* business application systems in the form of Enterprise Resource Planning (ERP Systems), Customer Relationship Management (CRM Systems) and Production Planning (PPS). This software category is the focus of the growing academic field of Process Mining (van der Aalst, 2011), which has been providing analytics for the understanding and improvement of these core business processes for many years.

In recent years, however, complementary software for *employee collaboration* has been put into the spotlight by the COVID-19 pandemic (Bullinger-Hoffmann et al., 2021; Richter, 2020). Enterprise Collaboration Systems (ECS) provide the digital support for the *non-process aware, communication and ad hoc oriented work activities* for communication, cooperation, coordination and the joint work around content (Koceska & Koceski, 2020). Virtually overnight, companies had to provide their employees with the necessary digital support to work from home (Ferreira et al., 2020; Williams & Grams, 2022). This situation, where organisations are left with little choice but to provide new software without proper time to plan has been called "forced adoption" by Schwade and Richter (2022). The somehow hastened selection and introduction of collaboration software has led to a situation, where many user organisations have made a portfolio of software available to their employees but have, however, neglected to accompany the introduction with the necessary adoption-supporting measures (Alberts et al., 2023). As a consequence, many technology portfolios now consist of overlapping, redundant functionality (e. g. multiple tools for video conferencing, chat and wikis) and there is a lack of agreed practices regarding the use of these new tools (Schubert & Williams, 2022). This leaves users struggling with the choice of tools for certain use cases (Mosen et al., 2024) and adoption has been sluggish and slower than hoped for (Greeven & Williams, 2016).

After the pandemic, the advantages of some of the introduced changes became apparent and the need to support *distributed forms of work* remained high (Williams & Grams, 2022). As a consequence, collaboration software with a focus on *synchronous communication* has become a total necessity (most prominently video conferencing). The tools for other, asynchronous organisational work processes, however, such as *coordination* or the *joint work on content* were never properly introduced and their adoption is still lagging behind despite their obvious advantages (Schoch et al., 2023). The fast and in some cases hastened introduction of collaboration software now calls for a rethinking of the portfolios of tools and a purposeful, long-term building of a performant Digital Workplace (Williams & Schubert, 2018). For the development of future plans, it would be helpful (and we argue necessary) to know, how the available collaboration software products are *currently used*. For proper planning, decision makers need information about actual system use, employee interactions, amount and type of documents created, supported work practices, and so forth. Unfortunately, the analytical features of the

leading collaboration tools (MS 365, Atlassian Confluence/Jira, HCL Connections/Sametime) only provide simple usage statistics, if any at all (Schwade, 2021).

The academic discipline concerned with the analysis of data that accrues from the use of collaboration software (event logs, content data and organisational data) is called *Social Collaboration Analytics* (SCA) (Schwade & Schubert, 2017). The increasing interest in SCA is reflected in a growing research stream concerned with analysing and visualising collaboration based on data from collaboration software. To date, SCA has been primarily used for computing *metrics* describing system use, characterising users (Hacker & Riemer, 2020; Schwade & Schubert, 2019), analysing particular constellations of collaboration such as collaboration across hierarchies (Riemer et al., 2015) and analysing the structure of content in these systems (Williams et al., 2020). Yet, despite some efforts (Drodt & Reuther, 2019; van der Aalst, 2005), methods for the analysis of *sequences of activities* in *collaboration processes* have not been addressed to a large extent (Schwade, 2021). Studies on the current status quo of collaboration analytics have shown that user organisations cannot plan and improve the current technology landscape without the knowledge to what extent and in what way the existing software *has already been adopted* (Schwade & Schubert, 2018a). We thus argue that there is a need for a more nuanced understanding of how the existing technology is used, in particular in terms of the *collaborative work processes* of employees.

Since *Process Mining* provides us with methods and techniques for the analysis of processes, it suggests itself for the application also to collaboration processes. Previous research, however, shows that the analysis of collaboration processes by means of Process Mining is very difficult (Drodt & Reuther, 2019; van der Aalst, 2005). In this article we discuss the (multiple) challenges and present a novel approach that can help overcome some of the problems. We introduce *"Social Process Mining"* as a new method that encompasses the whole process from the extraction of data from the involved software products to the discovery of process models. We end with a demonstration of the resemblance between the resulting SPM models and the real-world practices of the *computer-supported cooperative work* that is recorded in the event logs.

1.1 Research aims and challenges

The study presented in this article focuses on the *analysis of the digital traces laid down in the event logs of Enterprise Collaboration Systems*. Our *research aim* is to explore the (mis)match between real-world processes and event data in Enterprise Collaboration Systems. The successful application of *Social Process Mining (SPM)* hinges on the possibilities to extract, organise and analyse ECS event log data and to make it available in a form that can be used to visualise and display collaborative work activities and provide insights into the collaborative work processes carried out in such systems. However, as we explain in the following, examining the event logs of ECS is non-trivial and requires the development of novel methods and computational techniques to make the data available, understandable and interpretable.

Whilst there is growing research interest in the analysis of digital trace data in information systems (Franzoi & Grisold, 2023; Pentland et al., 2020), there are, to date, few in-depth research studies that examine digital traces from the logfiles of large-scale, operational *Enterprise Collaboration Systems*. Researchers and data analysts who are performing studies on ECS event logs are currently limited by two challenges:

1. *Data challenge – preparing high-quality data (analysis **input**)*:
   Gaining access to the native event and content data from organisations' operational ECS and the subsequent creation of rich event and content research data that is suitable for Process Mining hinges on established and trusted relationships with user organisations and the availability of the necessary log processor software for data transformation.
2. *Description challenge – making sense of the PM results (analysis **output**)*:
   Once the rich event and content data has been generated, there is a need for suitable methods and techniques for the analysis of this data and the description and visualisation of the discovered collaboration processes.

For our research, we had access to a large operational ECS run by a Software-as-a-Service provider (the ECS Management Team) that entrusted us with the event data for this research. In a multi-cycle *Action Design Research*, we first extracted and analysed the data from the ECS. Based on the findings, we developed a method for data transformation and analysis. It was essential for our work that we used the digital traces of the people involved in this project to develop the methods. Analysing the digital traces of collaborative work processes requires an intimate knowledge of the underlying activity. In a multi-cyclical research process, we tested and incrementally evaluated the methods. Our work addresses both challenges outlined above and is focussed on (1) *data preparation* (especially extraction, transformation and enrichment) as well as (2) the *description of the (digitally-supported) collaboration processes* in natural language. With the help of the new Social Process Mining method and developed tools, we are now able to explore and visualise process models for collaborative work processes and compare the real-world processes with the computationally explored processes. It is our long-term goal to be able to compare the AS-IS processes with TO-BE processes and to identify possibilities for improvement.

1.2 Enterprise Collaboration Systems (ECS)

Enterprise Collaboration Systems are integrated systems that combine a range of collaboration software features (or "apps") through a uniform graphical user interface. Typically, they combine traditional groupware features (e.g. shared calendar, task management, file sharing) with features of Enterprise Social Software (e.g. wikis, blogs, forums) and Enterprise Social Media (e.g. social profiles, posts, comments, likes, reactions) (Schubert & Williams, 2022). ECS are information infrastructures (de Reuver et al., 2017) typically implemented by medium to large organisations to provide the functionality and technical infrastructure to support employee collaboration, the coordination of digital work and the creation and management of digital work products (Leonardi et al., 2013; Williams & Schubert, 2018).

However, unlike ERP systems, which are designed to support *high volumes of repetitive tasks and highly structured business processes* (e.g. order processing, inventory management), ECS are designed to support *ad hoc, less structured collaborative work activities* (e.g. project coordination, event planning). Whilst most ECS functionalities are available to every user, they are used differently by different people and workgroups. This is partly due to the fact that ECS adoption in organisations is frequently bottom-up (Richter & Riemer, 2013) and users and workgroups are provided with an empty system or workspace where they are free to choose how to use it (Nitschke & Williams, 2018). For example, one workgroup might use a wiki for knowledge management and a task board for the coordination of tasks, whereas another workgroup uses the wiki only for meeting minutes and does not use a task board at all (Nitschke et al., 2020). This

becomes even more variable when we examine the specific sequences of tasks being undertaken and the related content being created (Mosen et al., 2024). The less-structured, inherently malleable nature of ECS presents a significant challenge, making tracing and analysis of tasks and activities in ECS much more difficult than for structured, process-aware systems such as ERP systems. As a consequence, this may explain why there has (to date) been limited success in providing in-depth, data-rich analyses of the specific mechanisms, strategies and actions that are shaping ECS and transforming collaborative work practices.

In most organisations, the Digital Workplace comprises a *portfolio of collaboration software*, that provides registered users with a wide range of tools to support their collaborative work (Schubert & Williams, 2022).

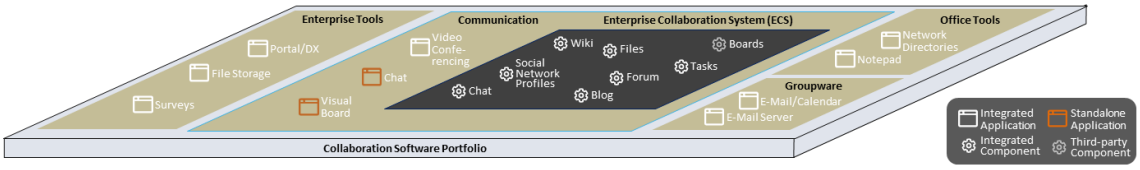


Fig. 1: Portfolio of collaboration software (adopted from Schubert & Williams, 2022)

Fig. 1 shows the conceptualisation of such a portfolio, in which the applications are grouped by their functionality into *Enterprise Tools* (portal/digital experience, central file storage and surveys), (near)synchronous support for *Communication* (video conferencing, chat, visual boards), *Office Tools* (notepad, network directories), basic *Groupware* tools (e-mail and calendar) and a *core ECS* with multiple components for joint work (e. g. workspaces that can contain chat, blog, forum, wiki, etc.). In this article, we are going to focus on the core ECS (dark area in Fig. 1) that we used to develop our method.

1.3 Social Process Mining (SPM)

In recent years, *Process Mining* (PM) has gained increasing importance in research and has become essential for business process management initiatives in companies (vom Brocke et al., 2021). The first (and according to van der Aalst probably the most challenging) step of PM is *process discovery*, which is used to generate process models that show the actual execution of business processes in business software, enabling companies to explore and understand their processes (van der Aalst, 2018, 2022a). As mentioned before, PM has been successfully applied to *process-aware* systems (PAIS), which "support processes and not just isolated activities" (van der Aalst, 2016, p. 27). PAIS support highly structured processes and are tightly linked to business processes, making their event logs ideal for applying PM.

Despite the obvious potential value for researchers and practitioners, only a few studies describe the application of PM based on logs from *collaboration* systems. In an early study from 2005, van der Aalst explored the application of PM in different Computer-Supported Cooperative Work (CSCW) systems and discovered several challenges (van der Aalst, 2005). The logs of collaboration systems are *unlabelled* as they do not contain a CaseID, which is required for PM. It was also identified that those logs are too fine-granular, leading discovery algorithms to create spaghetti-like process models that are hard to interpret and do not provide value for researchers and practitioners (van der Aalst, 2005). As highlighted, ECS support highly unstructured and flexible ad hoc collaboration through *fine-granular actions* such as creating a blog post or adding

comments and recommendations to such a post. While such actions are likely to be carried out as part of actual business or collaboration processes, ECS are not linked to business processes. To date, no solutions exist to make PM applicable in ECS, which means that there is a considerable and increasing blind spot in PM initiatives. Our research confirmed that the application of existing discovery algorithms to ECS logs results in uninterpretable spaghetti-models (cf. Section 4.1).

In this article, we introduce *Social Process Mining (SPM),* a novel method that aims to mine processes from the event logs of *Social Software* (Schwade & Schubert, 2017). The term "social" refers to the *functionality* that allows users to establish relationships between social profiles, e. g. by following or linking profiles and to interact around social documents, e. g. by editing, commenting, reacting or tagging (and thus extending) content created by other users. In this broad understanding, SPM can be used to analyse data from all Social Software, *corporate* Enterprise Social Software (ESS) as well as *public* Social Media.

The Social Process Mining discussed in this article uses state-of-the-art data science methods to discover, analyse and improve the sequences of collaborative work tasks. We aim to develop and provide the necessary techniques for the discovery of process models and the extraction of collaboration process patterns for the identification of current work routines and best practices, functional software limitations of collaboration software and in the long run also the detection of violations of rules and regulations.

The article is organised as follows: In the next section, we discuss related work, which we used to identify the current state of SPM and from which we extracted evidence for the fundamental challenges. In Section 3, we present the selected research design (Action Design Research, ADR), the involved participants and data sources. Section 4 is the main part, describing the detailed research activity and findings from the research project. The last section concludes our work and provides an outlook on future work.

2 Related Work

At the beginning of the ADR project, we performed a series of structured literature reviews for the thematic challenge areas. The findings were published in (Schwade & Schubert, 2018a, 2018b) and (Drodt & Reuther, 2019). Regarding the ***data challenge***, the literature review showed that whilst there is a growing research interest in the analysis of digital trace data, to date only a moderate number of in-depth, longitudinal research studies exist that examine digital traces from the logfiles of *collaboration software portfolios*. Extant studies are typically limited to *specific* ECS functionality and only few authors properly describe their data sources and their data pre-processing (Schwade & Schubert, 2018b). No publications could be identified that use methods of Process Mining on ECS. The literature review regarding the ***explanation challenge*** revealed a similar picture. Whilst there is a multitude of classification schemes for collaborative work, they are all abstract and only some of them provide actual values or instantiations of the proposed concepts (Schubert, 2024a). Typical examples are approaches that aim to classify work situations according to dimensions (Schubert & Williams, 2022), among them *synchronicity* (synchronous/asynchronous), *place* of work (co-located/distributed), type of *group process* (communication, cooperation, coordination), *content type* (text, image, video, audio) and number of *communication partners* (one/many).

The following sections discuss related work that informed the description of the pivotal challenges, the formulation of our problem statement and the development of the basic concepts and terminology for the new SPM method.

2.1 Challenge 1: Data Challenge

The reasons for limited prior application of Process Mining to collaboration technologies relate, to a large extent, to two key difficulties: the availability of suitable research and *data collection methods* and, *data access* and data handling. Digital traces are non-reactive system data and descriptively thin (Janetzko, 2017). They are not recorded for the purpose of research and analysis but for the correct operation of the collaboration software. Thus, they need to be transformed and enriched in an intricate process to reach a level of quality that makes them useful for SPM.

Gaining access to organisational data. Gaining access to data from ECS is challenging in terms of both, *organisational approval* and *technical availability*. As data from operational ECS contains confidential and personally identifiable information, the *organisation's approval* is required for extraction and use. Data requirements and the parameters and scope of data usage must be clearly defined and agreed in advance, through research ethics and data management agreements. This process takes time and requires the building of strong relationships between the research team and the organisations involved. For our study, we were granted full access to *an operational on-premises* installation of HCL Connections (CNX) and its databases. It was therefore possible to extract and transform all necessary data. The ECS is operated by the ECS Management Team involved in this study, which means that the people carrying out the research produced some of the digital traces on the platform themselves. All involved persons have given their consent for their data to be used.

The *technical availability* of event data is often limited by the *operations model* of collaboration software. Whilst *on-premises solutions* usually provide full access on the database level (e. g. via ODBC), collaboration software that is provisioned in a *Software-as-a-Service* (SaaS) model might

provide limited or no access to the underlying databases and only offer access on the application level (through APIs defined by the software manufacturer or service provider). There are further *technical access challenges* as the log data is stored in heterogenous formats (van der Aalst, 2016; Vianna et al., 2019; Williams et al., 2020). Event data is also often only stored for a limited time before it is deleted (Schwade, 2021). The ECS Management team providing the data for this research had anticipated a future need and made sure that the logs were extracted and permanently stored, which allowed them to accumulate content and event data of more than eight years.

Format and recording of ECS log file. Event logs have various formats and ways of being recorded. In SPM, we call their structure *schema* since the term "format" implies information on the actual file (format) the data is stored in, rather than what the included attributes are. Web server logs, for example, are often stored in plain text for easy access in command lines, whilst others are stored in databases or specialised formats (e. g. XES) for subsequent use (van der Aalst, 2022b). The technical implementation (schema and database type) of event logs is very dissimilar in the commonly used collaboration products. To give a few examples, CNX uses a relational database (DB2; 16 attributes), the database type of MS 365 logs is unknown to the user and only accessible as a cloud service (Management Activity API; 54 attributes), the event log of Hyland Alfresco is stored in a relational database (PSQL; 15 attributes) and the ISW Huddo Boards event log is stored in a document database (MongoDB; 14 attributes). Another big player, Atlassian, does not even provide access to the event logs in the SaaS versions of Jira and Confluence. We assume that these differences exist because the (external) provision of event data is, to date, *not a requirement* nor is its schema standardised in any form. So far, access to this kind of data has not (yet) been demanded by user organisations, which might change once the need for analytics increases. Specialised tools for event logs are (virtually) non-existent in collaboration software and at best there are rudimentary analytics features, e. g. simple usage metrics (Schwade, 2021).

An excerpt of an event log, where a user uploaded a file to a community in *CNX*, is shown in Fig. 2. Each record has the following attributes: record identifier (ID), actor (USER_UUID), affected content item (ITEM_UUID), the used module (CONTENT_TYPE_ID), the event type (EVENT_OP_ID), the workspace in which the event happened (COMMUNITY_UUID), the timestamp (EVENT_TS) and nine additional attributes. As depicted in Fig. 2, the central event table contains primarily IDs, and the related values can be retrieved from dimension tables. When using CNX, uploading a file *automatically* triggers two follow-up events, which is a sign of the fine-granular nature of the ECS event log. The event data satisfies two of the three log requirements for Process Mining (van der Aalst et al., 2012), as an *activity* (EVENT_NAME) and a *timestamp* (EVENT_TS) (used to derive the order of the events) can easily be identified. Identifying the third requirement, a *CaseID*, is more challenging, as will be further discussed in Section 2.2.

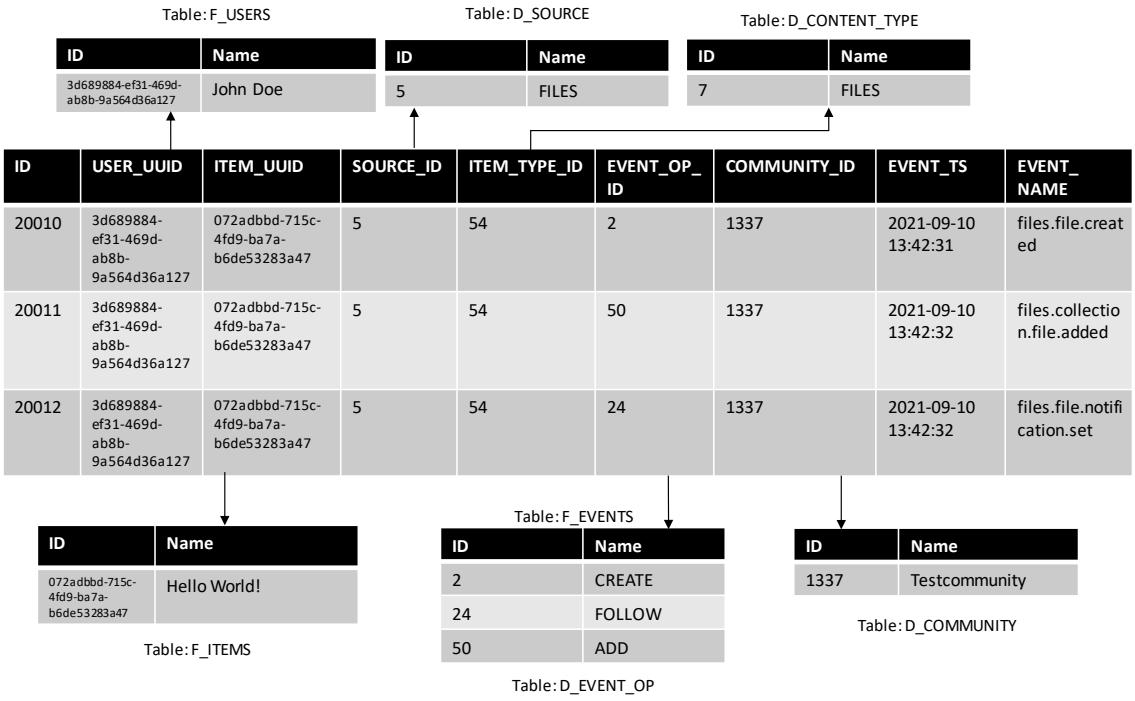


Fig. 2: CNX event data (excerpt)

Data quality. Collaboration software comes with several issues regarding the quality of its event data, which is a factor that "can prevent the identification of interesting patterns" (Schwade, 2021, p. 5). For a start, the schema of the log data chosen by the software developer is conceived for purposes other than research, and definitely not guided by the idea of using it for Process Mining. As a consequence, *data pre-processing* (cf. next subsection) is a sine qua non for the generation of suitable event data.

Digital traces in process-aware information systems (PAIS) are a suitable basis for research because recording them is a legal requirement (due to international accounting standards) and is necessary for technical reasons (roll-back). Thus, researchers can "piggyback" on this data for research. However, as there is no legal obligation to create logs for ECS, most of the systems record *incomplete event logs*. Logs are often limited to the tracking of *essential* changes to content (what was added and what is the current version).

Intrinsic event log issues relate to the log quality in terms of attribute completeness, event type completeness and log heterogeneity (Bose et al., 2013; Diamantini et al., 2014). While we aim to find all attributes of a comprehensive list (cf. C-Log) and reduce the complexity of (more) complex (e. g. mentioned, recommended, duplicated) operations to CRUD, this information is not always provided by the system. In ISW Huddo Boards, for example, the *creation* of a reaction is logged but the *deletion* is not. Whilst the activity names that are recorded in process-aware information systems (PAIS) are usually similar to the names of actual business tasks (e. g. "create bill", "issue invoice"), native collaboration software logs do not necessarily contain (useful) activity names. The attribute necessary to describe the software feature that was invoked by the user often needs to be artificially generated in the log pre-processing process (e. g. "create wiki page", "complete task"). The activity name reflects the type of content and the way it was manipulated. *Understanding* what the user was doing in that moment requires a translation of the software

feature into a *work task* (e.g. "create wiki page" → "prepare a meeting" or "complete task" → "task management"). This means that the problem described by van Zelst et al. (2021) of "events versus activities" is not the end of the discussion in collaboration software. Like in PAIS, we need to abstract fine-granular events to coarse-granular events (enriching the event record with the necessary activity names) to mitigate heterogeneity. In collaboration software, however, there is an additional step necessary to get to the description of activities at the business level (van Zelst et al., 2021, p. 722). We need a meta-construct, which combines multiple coarse-granular events (combinations of events or in Process Mining language *process patterns*) in a way that they describe task-level activities (scenes; cf. Fig. 3). These scenes can be used to determine the sequence of work tasks in a process model.

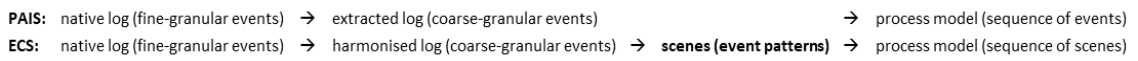


Fig. 3: PAIS log vs ECS log

Multiple systems view. As mentioned above, there are often *multiple systems involved* (Diba et al., 2020) in a collaboration process. For example, two colleagues might initiate a chat in Skype, create a wiki page containing meeting minutes in CNX, @mention a third colleague to loop him in and then plan the discussed tasks in Atlassian Jira. The actions of these three users all relate to the same "collaboration process" and therefore must be tracked across these three different systems. This means that access to the data must be established for each individual system and system-specific pre-processing steps must be carried out in order to obtain processes that span multiple systems. Without the inclusion of several systems, a holistic analysis of collaborative work is not possible.

Data Preparation for SPM. The academic field of PM provides methods and tools for the analysis of event logs (van der Aalst, 2016). For PM to work successfully, event logs need to be formatted using a standardised format (e.g. XES), so that they can be processed by PM tools (van der Aalst, 2022b). A review of the accessibility and format of the event log of HCL CNX, one of the leading collaboration software products, showed how difficult it is to create a uniform coarse event log that contains descriptive attributes (activity names) of the system functionality that is recorded in the event log (Just et al., 2024). This is a challenge for the analysis of collaborative work since collaboration processes frequently span multiple software products, all with their own proprietary log format. This means that we need a *data extraction and preparation pipeline* for multiple source systems that is suitable for SPM to extract, enrich (*incorporating additional sources*, namely the content databases) and transform (e.g. abstract and combine) the data to ensure a sufficiently-high data quality. To "enrich" means that the target schema includes the attributes necessary for later analysis (e.g. responding to "W-questions" such as "who did what when where and with whom"?). This augmentation and the following harmonisation are necessary because the number and characteristics of the attributes in the native event logs are very diverse.

Data pre-processing. To address the data quality issues of existing event logs in collaboration software the *Data Preprocessing for Cross-System Analysis* (DaProXSA) approach by Just et al. (2024) was chosen. This novel method for the *harmonisation and aggregation of log files* from collaboration systems is based on known approaches and frameworks for data mining, data pre-processing and Social Collaboration Analytics (SCA) (Chapman et al., 2000; Diba et al., 2020;

Fayyad et al., 1996; Schwade, 2021). The SCA framework was developed "for establishing the analysis of collaboration activities in the digital workplace" (Schwade, 2021, p. 1) and serves as part of the foundation of DaProXSA. In this approach, events are described as "user actions on documents" (Just & Schubert, 2023). The concept is formalised in ColActDOnt. The ontology specifies the concepts and properties of collaboration events.

2.2 Challenge 2: Description Challenge

Recent studies of *Social Process Mining* show that automated *Process Mining* methods, which usually consider structured processes, are not yet suitable for *Social Collaboration Analytics* due to the unstructured sequences of user activity that occur in ECS (Blatt et al., 2023; Drodt & Reuther, 2019). To date, no common standard or terminology for event logs in ECS exists, which makes it challenging to apply fully automated analysis methods and implies the need for extensive pre-processing to combine, harmonise (Just & Schubert, 2023) and abstract (Blatt et al., 2023) the data to improve interpretability (van Zelst et al., 2021). Further studies in the fields of *Process Mining* (Biuk-Aghai et al., 2005; Hartl et al., 2023), *Routine Dynamics* (Budner et al., 2022) and *Computer-Supported Cooperative Work* (Arazy et al., 2020) all report similar challenges with the identification of collaborative user activity from event logs with fully automated methods. The essential analytical hurdle is to translate and abstract the event logs generated in ECS into meaningful descriptions of the collaborative user activity to better understand actual collaborative work practices in digital workspaces. To date, few in-depth empirical studies provide technical insight into the harmonisation of digital traces from ECS event logs (Just & Schubert, 2023). Even if such data can be obtained in the form of rich homogeneous data records, there is still the *problem of description*, that is, "translating" events ("a user performing an action on a content item") into a meaningful description of the *type of work* or *work routine* that is represented by such fine-granular user activity (Pentland et al., 2020). Thus, new analysis methods and metrics are required to advance research in this area, which we address in the research described in this article.

Digital workspaces and work processes. A large percentage of the digitally-supported collaborative tasks in companies are carried out in *digital workspaces*, the digital environments where organisational units and project teams work together. Digital workspaces are created by selecting and assembling the required functional components provided by the collaboration platform. The creation of a new workspace starts when a workgroup is being formed. The functionality offered by current ECS is often broad and decisions need to be made about the choice of the software components to support the different types of collaborative work. The design of each new workspace is dependent on the specific context of use, the nature of the workgroup involved, the type of work being undertaken and the affordances of the available technologies (Gerbl & Williams, 2023).

Collaboration processes are not like ERP processes. The nature of work that is supported by *collaboration* software is significantly different from the work carried out in *process-aware* information systems (such as ERP or CRM systems), which support clearly structured, recurring business processes. The collaborative work is more flexible and less well-structured, in that the tasks may be carried out in unpredictable, changing sequences.

There are many established methods for the exploration of *business processes* based on the analysis of trace data from *process-aware* ERP systems (van der Aalst, 2016); ideally these Process

Mining methods could be adjusted and used to analyse the event logs from ECS to better understand *collaborative work* in digital workspaces (Pentland et al., 2020). Unlike process-aware systems, such as ERP or CRM systems, from which processes can be explored and analysed in the form of formalised models (e. g. using BPMN), ECS are not prescriptive regarding work activity; the same task can be accomplished using different software functionality or modules in varying order (Schubert, 2024b).

The nature of collaboration processes. To successfully address the *description challenge*, we need to establish a clear understanding of the term *"collaboration process"*. As mentioned earlier, whenever possible, we use the established terminology from PM to make sure that existing PM tools can be used for analysis. The problem with the description of processes or process snippets is that there is no generally accepted classification for collaboration processes. There is a multitude of classification schemes for abstract constructs, but only a few concrete taxonomies provide descriptive labels (Schubert et al., 2025). Collaborative work processes are composed of sequences of actions, some of which are *synchronous,* requiring employees to work together at the same time (e. g. using a video conferencing tool), whereas other tasks are performed *asynchronously* where work is conducted sequentially, with one employee working independently on a task then handing over the work products to another employee when their part is done. In addition, an individual employee might be a member of multiple workgroups and, as a consequence, multiple digital workspaces, and move between them over a working day.

In Fig. 4, we conceptualise the workday as a sequence of work scenes. Collaborative tasks can be carried out synchronously or asynchronously, in the private dataspace of the individual or in a space shared with colleagues and the activities can be digitally-supported or physical. It is important to bear in mind that not every step of a collaboration process is digital and thus recorded in an event log. The digital traces will never give us a complete picture and the resulting metamodel might be incomplete.

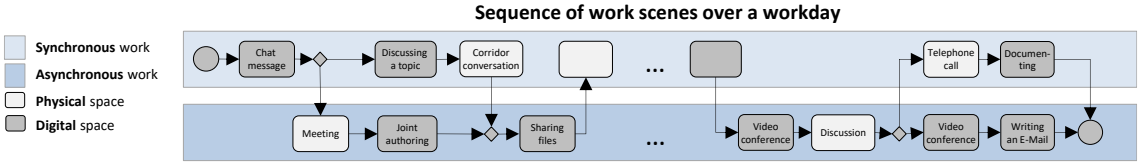


Fig. 4: Workday as a sequence of work scenes

PM was conceived for structured, repetitive *business processes*. Weske defines them as follows: "A business process consists of a set of activities that are performed in coordination in an organisational and technical environment. These activities jointly realize a business goal." (Weske 2007, p. 5)

Collaboration processes, on the other hand, are not well structured and whilst the effective sequence of activities follows certain *patterns,* each concrete process *instance* is usually *slightly different*. In 2005, van der Aalst investigated the possibilities of applying Process Mining to digitally-supported collaboration processes, or in his words "less structured processes supported by CSCW systems". In our research we are following up on these early investigations and deepening the understanding of the necessary trace data as well as the observable forms of collaboration processes.

In an attempt to stay close to the understanding of a business process in PM, we merge the definition by van der Aalst (2005) and Weske (2012) and define the term "collaborative work process" as follows:

A ***collaborative work process*** (shown in Fig. 5) consists of a sequence of work scenes (a set of activities involving two or more people) who are jointly working on a shared task. Such collaboration processes can be supported by one or more collaboration tools. A *work scene* occurs in a workspace and is defined by the involved actors. In the scene, the actors create one or more content items in a container. The digital support of collaborative work processes (with the aim of completing a task) frequently spans multiple software tools.

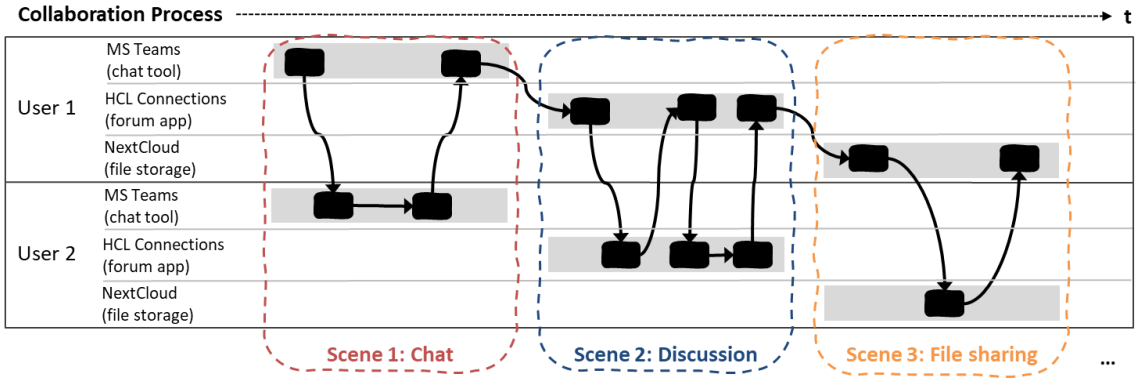


Fig. 5: Container bracketing: events of two users in three apps

To give an example, a process for the joint creation of a financial report might be initiated asynchronously in a chat, then be followed by a discussion in a forum app (board) and the participants will then usually continue to extend and finish the document asynchronously after the meeting ended (file sharing and editing). This means that such a collaboration process is characterised by handovers of work on documents and transfers between different software tools. The common understanding of a scene according to dictionaries is: "the place where some action or event occurs", "a *division* of a play[…], usually representing a *passage of time* in a single setting, *featuring* a specific character or *group of characters*", "a unit of action or a *segment of a story* in a play"[1]. In the context of SPM, a scene is a concept used to describe a *segment* of a collaboration process that contains a bundle of (inter)actions of a *group of actors* in a specific *location* (a container in a workspace) in a related *time frame*. This definition is in line with previous uses of the term e. g. by Simões et al. (2018) and Schön et al. (2019).

As can be seen in Fig. 5, a collaboration process is a sequence of consecutive scenes in different software systems. A single scene (e. g. a chat conversation or a joint discussion in a forum) can be computationally identified because it happens among a given group of actors, in a limited time interval, on a single document and in the same software tool. We extend the concept of temporal bracketing (Hartl et al., 2023) and perform *container bracketing*. We bundle the events in a specific container that occur in a specific time frame into a scene. The transfer takes place when the same group of actors starts their work on a new document in a different functional software module.

[1] Source: dictionary.com/browse/scene, accessed 12.07.2024

For the identification/description of collaborative work processes we need descriptive labels for the scenes. The combination/sequence of certain scenes is an indicator for a specific type of collaborative work process. The IRECS model (Schubert, 2024b) provides a taxonomy for ECS. Two of the IRECS levels (cf. Fig. 6) were identified as candidates for the description of collaborative work. The description of the concepts is available in an online repository[2].

Collaboration Scenarios	Administering documents Administering social profile Administering tasks Alerting Conducting a meeting	Conducting a poll/vote Conducting a survey Discussing a topic Documenting Enriching information	Following people/content Giving Feedback Joint authoring Posting a short message Preparing a meeting/event	Rating information Retrieving info/search Sharing files Sharing information Writing meeting minutes, …
Collaborative Features (C^4)	**Communication** Texting Chatting Visualising Voice calling Video conferencing Commenting, annotating Discussing, …	**Cooperation** Workspace sharing Shared authoring Screen sharing Markup changes User profiles/social profiles Ratings, rankings Workspace awareness, …	**Content Combination** Document management Content management Content collection Content subscription Aggregation/integration Tagging Surveying, …	**Coordination** User directories and roles Calendar planning Resource planning Shared tasks Reminders, triggers, alerts Workflow support Presence awareness, …

Fig. 6: IRECS taxonomy: collaborative scenarios (processes) and collaborative features (scenes)

Level 2 contains "collaboration scenarios", which are *collaboration processes* in the understanding of Process Mining. Level 4 contains "collaborative features", which describe the functionality of ECS and are natural candidates for the description of *work scenes*. We selected the IRECS taxonomy as our starting point for the development of descriptive labels for collaborative work. Table 1 shows examples of scenes with their typical container and examples of software products.

Table 1: Work scenes (labelled using the IRECS taxonomy) and their typical containers

Work scene	Container: content types	Example software products
Chatting	Channel: microblog post, comment, file	iMessage, WhatsApp, Skype, Teams
Visualising	Workspace: board, content elements	Miro, Mural
Video conferencing	Meeting: microblog post, audio transcript	Zoom, BBB, Sametime, Go2Meeting
Discussing a topic	Forum: post, comment	CNX Forum, IP.Board
Shared authoring	Workspace/folder: file, revision marks, comments	OneDrive, Nextcloud, Connections Docs, Google Files
Sharing information	Blog: post, comment	CNX Blog, WordPress
Sharing files	Folder: file, comment	CNX Files, Nextcloud, Box
Shared tasks	Board: todos, notes, comments	Huddo Boards, Atlassian Jira, Asana

Social Documents: The Content in Enterprise Collaboration Systems. Social documents are the output of the work of people in ECS (e.g. blog posts, wiki pages), and "are created as people collaborate on joint work" (Williams et al., 2020, p. 2826). The structure of social documents has

[2] w3id.org/CEIR/irecs

been formalised by Williams et al. (2020) in the Social Document Ontology (SocDOnt). Fig. 7 shows an excerpt of the essential concepts in SocDOnt.

Social documents are abstract objects, which are viewed as compositions of content items. The initial content item (root item) of a social document is the *intellectual entity*. These can be, for example, files, board posts, microblog posts, tasks, blog posts or wiki articles. Other *items*, also referred to as social document *components,* can be added (by other authors) to a social document. Components can be *intellectual components* (e.g. comments or attachments), which contain text or images provided by the author or *simple components* (e.g. tags or recommendations), which are "simple" reactions or classifying markers. *Containers* are high-level concepts and group items stored by the same application (e.g. forum, weblog, wiki, microblogs). Containers and social documents are stored in *spaces*. The (group)spaces are the virtual environments where groups work together.

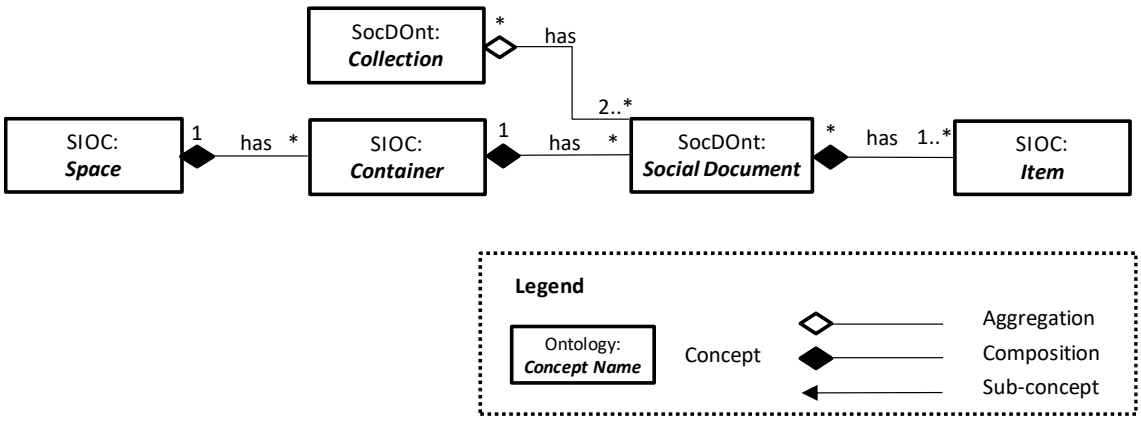


Fig. 7: Overview of core concepts in the Social Document Ontology (excerpt) (Williams et al., 2020)

Social documents as traces of collaborative work. Social documents are initiated by the *create action* of the user who creates the *intellectual entity*, which is the starting point (core) of the document. Once created, the intellectual entity (violet) can be enriched by further content items (yellow) by any author that has access to the document (cf. Fig. 8). This way, the items of a social document can be read (R), changed/updated (U) and deleted (D) and additional content elements (*components*) can be created (C), changed/updated (U) and deleted (D) by *multiple authors*. Social documents are compound documents and can contain *multiple different content types* (e.g. a forum contains posts with responses and tags).

Fig. 8: Sequence of content creation: growing document graph

The sequence of content creation can be visualised in a graph structure (cf. Fig. 9). The social document is an ideal study object for examining the joint interactions of people around specific content (Mosen et al., 2024).

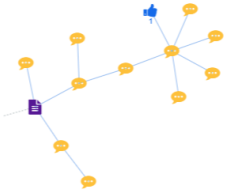


Fig. 9: Structural view of joint content creation (document graph) (Mosen et al., 2020)

3 Research Design: Action Design Research (ADR)

As discussed in the section on related work, conducting Social Process Mining is a multi-facetted task and we knew that we had to overcome multiple challenges on the way to a successful SPM Method. SPM is a practical domain and the aim of SPM is to help user organisations to better understand how the software is used. Against this background, we needed to work in a concrete organisational setting and closely with the actual users of SPM to identify their needs and to receive continuous feedback. Also, our development and testing required a suitable data source (containing events of the collaborative work of real people) for development and testing. We chose Action Design Research (ADR) as our research method because it was ideally suited to our needs. Sein et al. developed ADR as "a research method for generating prescriptive design knowledge through building and evaluating ensemble IT artefacts in an organisational setting" (Sein et al., 2011, p. 40). We were able to perfectly match the principles of ADR in our research setup. We worked closely with a provider of a hosted service for a large Enterprise Collaboration System and were thus able to develop and test SPM in this realistic organisational setting. This allowed us to perform multiple cycles of development and testing of methods and tools, to evaluate the results and use the findings to develop *design principles* for the next cycle. In doing so, our research accounts deeply for the context of the user organisation and incorporates the "to-be-developed artefact's organisational stakeholders and end-users in the research process" (Peffers et al., 2018, p. 134).

3.1 Research Design

Fig. 10 shows the cycles of the ADR project. As suggested by Sein et al. (2011), each cycle contained the stages of problem formulation, building (plus intervention and evaluation), reflection (and learning) and formalisation of learning (output). At the time of writing this article, the ADR project had been running for 6 years and had gone through three major cycles. The current *SPM Cockpit* is a functional prototype and addresses all of the discussed challenges in a proof of concept.

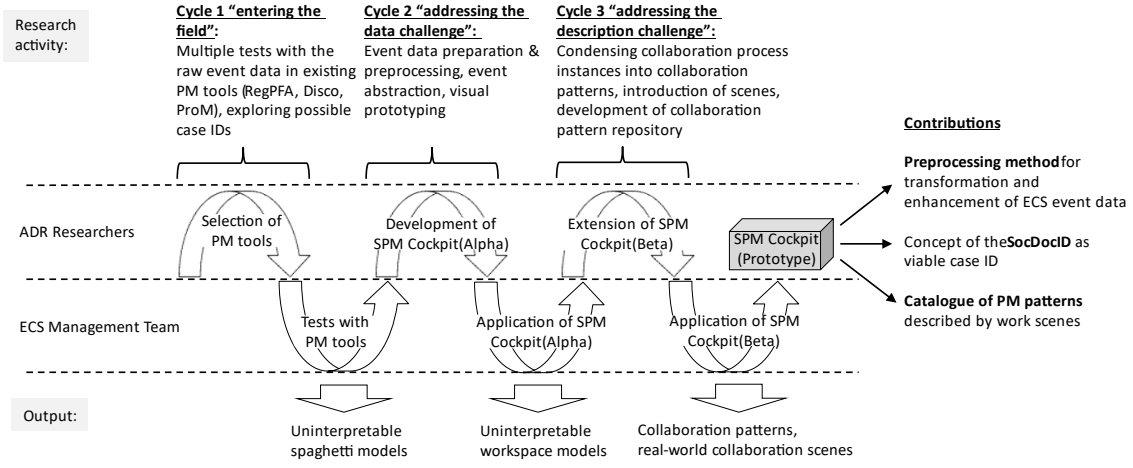


Fig. 10: Cycles of the Action Design Research following Sein et al. (2011)

Cycle 1. "Entering the field". The first phase of the ADR project was characterised by multiple tests with the raw data using *existing PM Tools* (RegPFA, Disco, ProM) aiming at finding a suitable CaseID. The results were uninterpretable spaghetti models (Drodt & Reuther, 2019). The main

finding from this phase (which served as an input for the next phase) was that ECS logs cannot be used in their native form and have to be substantially pre-processed.

Cycle 2. "Working on the data challenge". The focus in this phase was on data preparation. An observer was created to create a reliable high-level log. With the help of the observer log, an algorithm for event log abstraction of the low-level log was developed (Blatt et al., 2023). An alpha version of a specialised software artefact was created (*SPM Cockpit Alpha*). The results of the app were uninterpretable workspace models. The output of this phase (and thus the input for the next phase) was the realisation that we need the social document ID (SocDocID) as a CaseID because joint work happens around documents.

Cycle 3. "Working on the description challenge". The focus in this phase was on the use of the SocDocID as a CaseID. The SPM application was further developed into a beta version (*SPM Cockpit Beta*). A framework for the description of the collaboration process patterns was developed and published (Schubert, 2024b; Schubert et al., 2025). The next and final step in this project was then the (further) development (and making publicly available) of a *catalogue of collaboration patterns* (models) and their *corresponding scenes* (feature bundles).

3.2 Participants: ECS Management Team and ADR Researchers

The user organisation which provided the context and the data for the ADR research project is the University Competence Center for Collaboration Technology (UCT), a provider of a large-scale Enterprise Collaboration System, which we call the ECS Management Team (Schubert & Williams, 2016). The ECS Management Team runs an instance of CNX as a Software-as-a-Service (SaaS) for educational institutions. At the time of writing this article (2024), the ECS had been in operation for 10 years. The ADR research project had been started six years earlier (in 2018), when the *ECS Management Team* had approached the *ADR Researchers*, a team of academics with a background in Process Mining and Enterprise Collaboration Systems, with the request to analyse the use of the ECS.

3.3 Data Source: UniConnect

The ECS used for this research is an operational instance of CNX[3], which provides multiple collaboration software modules including microblogs, blogs, forums, wikis, files and task boards. The platform *(UniConnect)* has 3,718 activated user accounts, 2,430 workspaces and 5,612,823 event records (unfiltered, for the time period from 2017 to 2023).

[3] hcl-software.com/connections (CNX)

4 Development of the SPM Method: Design Science Cycles

In order to design the SPM Method, we conducted three design cycles as described in the research design. At the end of each cycle, we reviewed (and published) the intermediate results, which helped us improve and deepen our understanding of the ECS event data, make important adjustments for the pre-processing of the event data and develop design principles for the digital artefact (the *SPM Cockpit*).

4.1 Cycle 1: "Entering the Field"– Experimenting with the Raw ECS Event Data

Project activity. We started the ADR project with a series of workshops in which the UCT Management Team assumed the role of the *Client* in need of platform analytics, which the ADR Researchers in the role of *IT Consultants* were meant to provide. The workshops helped to clarify the project goals, to develop a joint language (terminology) and to discuss possible ways to use Process Mining techniques for the analysis of event data. It was decided to use the terms defined by the *Collaborative Actions on Documents Ontology* (ColActDOnt) for the description of the theoretical concepts. A plan for the data extraction was developed and a preliminary data set was jointly extracted from UniConnect. The ADR Researchers began the investigation with an explorative analysis of the raw event log using process discovery methods in existing PM tools (Drodt & Reuther, 2019).

Research findings. The ADR Researchers transformed the native event records from the database into XES (using the user as CaseID), applied trivial filters to exclude read and visit events and tried to explore user activities and their sequential relations. The initial results were "spaghetti"-like process models. In the next step, we tried to predict user interactions using RegPFA (Breuker et al., 2016), a predictive process discovery algorithm. The tests showed that the log contained too many raw system events, which made it impossible to produce a suitable model that provides a basis for a meaningful analysis of collaboration processes in ECS.

In further experiments, the ADR Researchers utilised ECS event data to discover different views on possible collaboration processes, focusing on finding an attribute that is a suitable CaseID. The Fuzzy Miner, implemented in the PM tool Disco[4] is suited for handling unstructured log data and abstracting complex process models (Günther & Rozinat, 2012; Günther & van der Aalst, 2007; Rozinat, 2013), making it a suitable choice for SPM. We tested the three most promising attributes, Account (USER_ID) Item (ITEM_UUID) and Space (COMMUNITY_ID) as CaseIDs and tried interpreting the resulting models. Table 2 provides an overview of the selected IDs in the event log and an assessment of their suitability for use as a CaseID.

[4] fluxicon.com/disco

Table 2: Summary of evaluation of using different CaseIDs

ID	Implications on process models	Characteristics of resulting process models
Account (USER_ID)	When selecting the Account as a CaseID, the traces represent individual users' actions. As traces are constituted for individual users, the process models do not provide insights on collaboration patterns between multiple users.	The resulting process model typically contained very long paths.
Item (ITEM_UUID)	The resulting traces are constituted for each social document *component*, showing the different actions performed on them. Considering a blog post, adding a comment to the original post was considered in a separate trace because the blog post and the comment have different ITEM_UUIDs.	As the ITEM_UUID in CNX is very fine-granular, the resulting process models contained many short paths consisting of only one activity. This is not surprising because many content components allow only one action to be performed.
Space (COMMUNITY_ID)	In this case, each trace represents a workspace. It could be assumed that the resulting process model would allow to identify typical collaboration patterns in communities. However, we observed several limitations in this process model.	First, we observed that the process model often contains very few paths consisting of a high number of activities. These very long paths are caused by activities that are unrelated as they were executed in parallel but depicted as sequences in the process models.

As shown in Table 2, the event log of CNX does not contain a suitable CaseID. In the following, we present exemplary results from using the COMMUNITY_ID and the ITEM_UUID as CaseID. We extracted the event log from UniConnect for the year 2021. The event log contained 1,122,167 events generated by 1,174 users across 633 workspaces involving 234 unique activities. The high number of activities recorded in the log is a first indicator for its fine granularity. When set to show 100% of activities and 100% of paths, Disco returned the unfiltered process models shown in Fig. 11 and Fig. 12, the first with the COMMUNITY_ID and the second with the ITEM_UUID as CaseID. A closer inspection of the process model revealed that the complexity mainly stems from the very high number of variants, making it impossible for analysts to interpret such a complex process model. The high number of variants is caused by the high number of different activities and the overall number of events per case.

Fig. 11: Unfiltered Process Model (CaseID: COMMUNITY_ID)

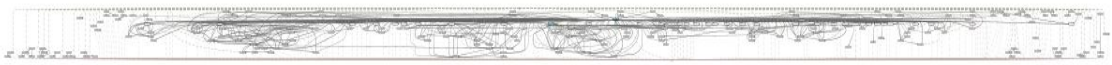


Fig. 12: Unfiltered Process Model (CaseID: ITEM_UUID)

The process model became more readable when reducing the number of possible paths. Fig. 13 shows a filtered version of Fig. 11 (CaseID: COMMUNITY_ID) with only 1% of the paths, which still shows a very complex model with many variations. As can be seen, the reduction of paths does not help in terms of interpretability and the loss of information is not acceptable for a holistic analysis of collaborative work activity anyway.

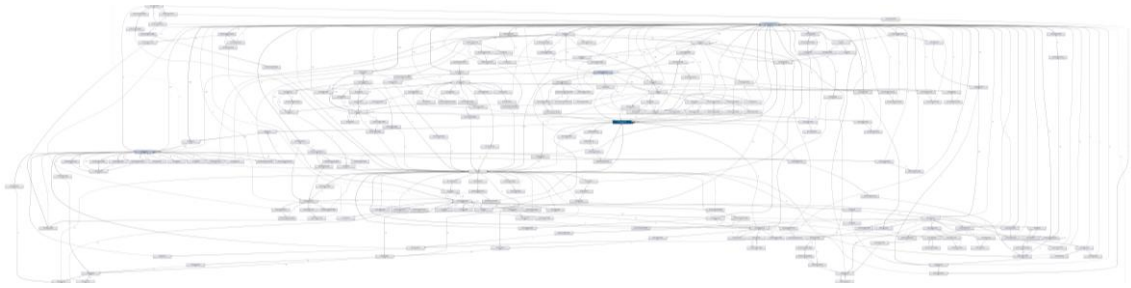


Fig. 13: Filtered Process Model (CaseID: COMMUNITY_ID; filter 1% of paths)

In the next step, we filtered the process model to show only event data from *one* particular workspace. The resulting process model (Fig. 14) shows 100% activities and 100% paths. As summarised in Table 2, the process model contains many short paths consisting of one or only a few activities, which is an indication that we see typical sequences of system functionality rather than meaningful collaboration processes. Closer inspection confirmed this suspicion. CNX records an excessive number of events corresponding to “read” and “visit” events. A particular issue with read and visit events in CNX is that they are triggered after almost every user action. For example, after creating a blog post, the user is redirected to this blog post. The log triggers a read event for this post associated with the user ID (another indicator for the fine granularity of the log). Consequently, the related activities take a central role in the process model, resulting in many loops that interrupt longer process paths. Even with state-of-the-art software, it is currently impossible to automatically filter out the system-generated events. They cannot be distinguished from regular read or visit events resulting from actual and deliberate content views.

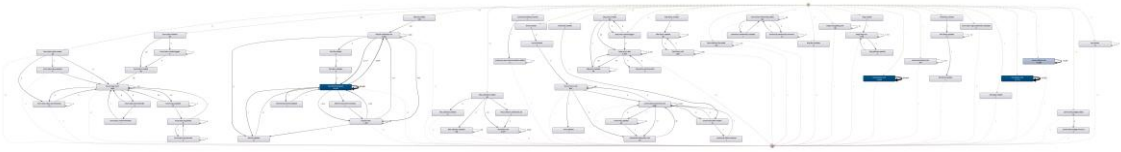


Fig. 14: Process Model for a selected workspace (CaseID: ITEM_UUID)

To address this problem, we followed the suggestions by Droth and Reuther (2019) and filtered all read and visit events from the event log in the next step of our exploration (in acknowledgement of the resulting potential information loss). The resulting process model is shown in Fig. 15 and has become much more readable. With the filtered read and visit events, most paths became slightly longer. Furthermore, most of the paths of the process map seem to be independent branches as they finish with the end event and have no transitions to other paths. A closer inspection of these branches reveals that they are separated by the available modules (ColActDOnt: containers) in the workspace, i. e. sub-processes in the *forum* are separated from the sub-processes in the *blog* of the workspace. This observation is not surprising and a logical consequence and limitation of choosing the ITEM_UUID as a CaseID.

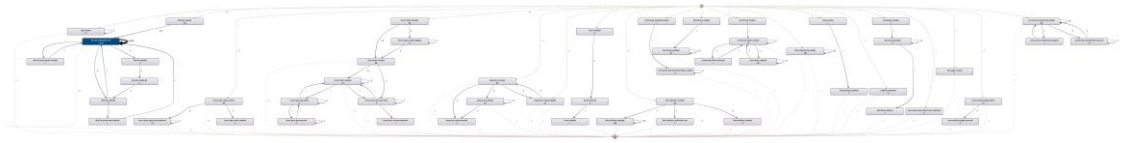


Fig. 15: Process Model for a selected workspace with filtered events (CaseID: ITEM_UUID)

Now that we had established that a process path is built around the actions on a *certain type* of content, we performed a final step and filtered the event log to only show events from a single container, i. e. the *forum* of this workspace. The resulting process model is shown in Fig. 16. With the process model being filtered to the forum of the particular workspace, several paths become clearly visible that can possibly be interpreted. The sub-process on the right, for example, shows the process of creating topics (a discussion among people) in the workspace.

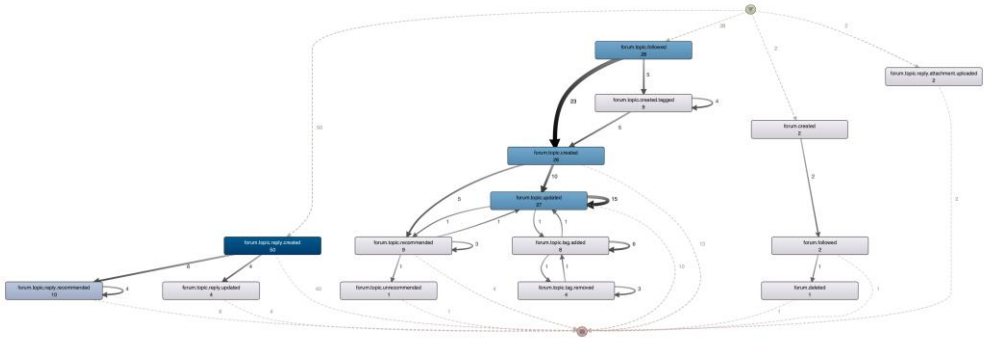


Fig. 16: Process Model for the forum in a selected workspace (CaseID: ITEM_UUID)

Here, another issue with the low-level log of CNX can be observed. The sub-process starts with the event *forum.topic.followed,* which is followed by *forum.topic.created*. When a user creates a topic in the forum, the user automatically subscribes and thus follows the topic. In the event log,

this is represented by the sequence consisting of these two events, which are both associated with the actual user ID. When analysing the low-level event log of CNX, we observed that these two events occurred in different orders for the same user action. Such event sequences that are triggered automatically are problematic for SPM because they do not reflect actual and deliberate user actions. Thus, in many cases, they lead to misleading process models. The paths with the highest frequency following the *forum.topic.created* activity node leads to the *forum.topic.updated* node, which shows that in 50% of the cases, when a forum topic was created, it was immediately updated, for example, to correct errors or add information. It also appears that some forum topics were immediately deleted after their creation. Apart from these observations that indicate an individual's behaviour for content maintenance, this sub-process does not allow the discovery of *collaborative* processes *between* individuals. The process model merely describes the lifecycle of a forum topic and the order in which the system allows the execution of particular actions.

It is noteworthy that there are no connections between these two sub-processes in Fig. 16. It could have been expected that the parts of the process for creating a forum topic and replying to a forum topic are connected. As mentioned before, the reason for the many isolated sub-processes is a result of choosing the ITEM_UUID as the CaseID. In the forum, the topics and their replies have different ITEM_UUIDs, which is why they are depicted as separate processes in the process model. Furthermore, the knowledge gained from the process model is limited. It is likely that the models merely show system functionality and cannot help us to show actual user behaviour. Also, it remains unclear for which purpose or as part of which process an action was executed. Another limitation of the process models generated by Disco is that they do not show interactions between users (collaboration). In the process models, an event sequence initiated by one user looks the same as the same event sequence by multiple users. As argued throughout this article, the native event logs of ECS are on a rather fine-granular level, which is not suited for SPM.

To summarise, our tests showed the following challenges with the low-level log of CNX: One user action in the system may result in multiple low-level log events, which may also occur in different order for the same user action. Consequently, the low-level events of two or more user actions may overlap, making it impossible to apply simple mapping techniques. We also observed that the low-level log contains events where no user activity was performed. In the Process Mining literature, such fine-granular logs are also referred to as low-level event logs. In low-level logs, it is often impossible to associate fine-granular low-level events with the corresponding user activity (Bose et al., 2013; Bose & van der Aalst, 2009; Zerbato et al., 2021).

The purpose of the first ADR cycle was to test the feasibility of using the raw data with existing PM tools. We extracted three major insights as design principles for the following cycles. (1) An ECS (in our case CNX but we found the same for other ECS) produces system events, which are not helpful for the analysis of actions between users (collaboration), (2) the available event attributes do not provide a viable CaseID and (3) events are recorded on a (too) low level of abstraction.

These insights led us to the second cycle, in which we focused on data preparation, *enriching* the event records (by artificially introducing a social document ID (SocDocID)) and *pre-processing* (filtering and abstracting) the data with the help of a novel method (DaProXSA).

4.2 Cycle 2: "Addressing the Data Challenge" – Enriching the Event Log

Project activity. The activity in this cycle focussed on addressing the data challenge. The ADR Researchers developed and applied a novel method for data pre-processing (DaProXSA). In parallel, the first alpha version of the *SPM Cockpit* was developed. This App is meant to assist the UCT Management (the client) with their desired data analyses.

Research findings. Based on the insights gained in the first ADR cycle and a complementary literature analysis on event data preparation, the ADR Researchers developed DaProXSA (Data Preprocessing for Cross-System Analysis, cf. Section 2.1) as a method for data pre-processing (Just et al., 2024). DaProXSA consists of seven data pre-processing steps for event logs from ECS, which mitigate the following data challenges described before: (1) heterogeneous log formats, (2) different levels of granularity, (3) missing CaseID and (4) missing attributes. We used the recommended methods for *data augmentation* and *abstraction techniques* to obtain an interpretable, high-quality event log for SPM. As a result, the final event log schema was based on the harmonised C-Log schema (Just & Schubert, 2023) and includes additional attributes from *organisational* data (e.g. user accounts and roles) and *content* data (e.g. wiki page names and wiki page content).

Selection, Flattening & Enriching, Extraction, Filtering. In the *collection phase* the available records were identified and *selected* from the system. The records were *flattened* (denormalised) and *enriched*, a process in which a number of additional attributes was added to the record to turn it into a C-Log. One of the *essential new attributes* is the *SocDocID*, a unique ID that is shared by all the items of a particular document on which the users collaborate. In the *cleaning phase* unnecessary log entries (e.g. redundant events or events that do not relate to content data) were *filtered*.

Social Documents as the key element for SPM. From the experiments with the use of different CaseIDs, we had ascertained that the ITEM_UUID produced the most meaningful process models. When people work together, they read and manipulate content items. One person creates an initial root item (e.g. a forum or blog post or a file) and others can then respond to this item (e.g. with comments or likes). With their response, they create new items that are linked to the root item (shown in the growing document graph in Fig. 8). So far, however, the process models for the ITEM_UUID are very short and do not show collaborative work due to the simple fact that, in the context of a blog post for example, the related comments have *differing item IDs* (in the native CNX event record) and are thus handled as independent cases, which does not help to discover interaction around a social document.

This is where the concept of the social document by Williams et al. (2020) provided us with a solution. Social documents can be represented as growing graphs (cf. Fig. 9), and they are composed of related items that share the same document ID. The existence of this shared social document ID (SocDocID) allows us to use it as an ideal CaseID. Unfortunately, this ID does not naturally occur in the logs of commercial collaboration software and has to be artificially generated and added to the events records during pre-processing. The introduction of the SocDocID that strings previously (programmatically) unrelated items together into a meaningful graph structure creates a new perspective that has previously not been available.

Event Abstraction. During the collaborative work of humans in an ECS, the system records low-level events. In order to conduct a suitable analysis using ECS events, we need to solve the issue of the low granularity of such events. For example, a high-level activity can consist of several low-level activities, which, in turn, can occur in different sequences or overlaps (cf. Cycle 1). Thus, the solution is the transformation of low-level events into high-level events, which is called *event abstraction* (van Zelst et al., 2021). In the *transformation phase* of DaProXSA, this *abstraction* is performed to reduce data complexity. This way, fine-granular events are bundled into higher-level events, which contributes to less complexity in the generated models.

Existing work has not been able to take the explained characteristics into account, which is why we needed to develop a suitable approach (Just et al., 2024). The idea is that we record high-level events for a predefined period of time. This is done using an *observer*, that recognises predefined user activities in ECS and records them with the needed level of abstraction. Then, using the native low-level events and the high-level events from the observer, we can, using a supervised machine learning approach, train an *abstraction model*. This model can then be used to abstract various low-level events to the desired level of abstraction. Thereby, irrelevant and duplicate events are removed. What remains are events that express the *actual intended collaborative user actions*. With the trained model, it is possible to abstract future events from other instances of the same software so that the training of the model only needs to be performed once per software product.

Data Harmonisation: Improved C-Log Schema. Introducing a common schema that harmonises the concepts and names of log records improves their readability and comparability. When comparing the native event log of CNX to the harmonised C-Log format (Fig. 17), we see that there are benefits in the (1) reduction of actions (in red) to CRUD operations and (2) in the omitting of module name (e. g. forum) because it leads to easier readability. Omitting the container does not lead to an information loss since there is (in most cases) only one possible intellectual entity type per container (e. g. wiki pages can only be found in wikis).

Native activity names in CNX	**Transformation to C-Log schema**
forum.topic.created	BoardPost.created
forum.topic.recommended	BoardPost.Like.created
forum.topic.tag.added	BoardPost.Tag.created
forum.topic.tag.added	BoardPost.Tag.created
forum.topic.tag.added	BoardPost.Tag.created
forum.reply.created	BoardPost.Comment.created
forum.reply.created	BoardPost.Comment.created
forum.reply.created	BoardPost.Comment.created
forum.topic.reply.recommended	BoardPost.Comment.Like.created
forum.topic.reply.recommended	BoardPost.Comment.Like.created

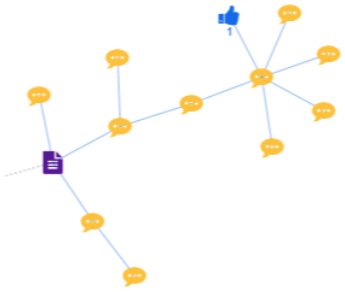


Fig. 17: Event log entries for growing document graph (left: native activity names, right: C-Log)

These modifications are essential when we track events over *several systems* because uniform names are now available for similar or even identical operations. A cross-system example from CNX and ISW Huddo Boards (IHB) is the following: The events *activity.entry.created* (CNX), *activity.entry.duplicated* (CNX), *activity.todo.created* (CNX) and *node.create* (IHB) are all transformed to *Task.CREATED*. Minor information loss is happening since the duplication of a task is not exactly the same as a creation, but the harmonisation helps to reduce variability in the

process models. Additionally, the *node.create* event from ISW Huddo Boards happens for different node types (boards, comments, entries, lists and tasks), meaning that the harmonised event name is in this case even more precise than the native system event name.

The ontology-based log schema introduced by Just and Schubert (2023) offers an object-centric representation of event data. All of the used concepts are modelled in the ColActDOnt. For the C-Log schema, selected data properties were chosen to gain a comprehensive list of attributes for ECS event log records. Some of these are necessary for SPM (in *italics*). For example, the *system instance* and the *social document ID* are essential for SPM. Without these, there is no CaseID and no way to differentiate the origin (system) of an event. The attribute list (cf. Table 3) was refined to also include a UTC timestamp which enables an *absolute ordering* of events. The two timestamps make it possible to analyse activities in relation to the local time (for example, whether different activities are carried out in the morning or the afternoon) and to have a complete chronological order of all events without additional data transformations (in BI tools).

Table 3: C-Log for SPM (further developed from Just et al., 2024, p. 8)

Attribute | Abbreviation	Example - HCL Connections	Example - ISW Huddo Boards
Agent Email	bobross@artcompany.com	bobross@artcompany.com
Agent ID (artificial)	f8819a77-7159-4b14-9196-acd25f513453	f8819a77-7159-4b14-9196-acd25f513453
Agent Name	Bob Ross	Bob Ross
Agent Type	Person	Person
Account ID	1bfb7393-d807-643f-bd26-18b752bca71a	622b0a7443607da298bbbf39
System Instance	ArtCompanyConnect	ArtCompanyHuBo
System Software Product	HCL Connections (CNX)	ISW Huddo Boards (IHB)
Space ID	e286f327-62be-43a9-9af3-ba7bb8af5a2c	6638bbe74ee49b57ef0a4701
Space Name	Creative Painting Space	Creative Painting Tasks
Space Type	GroupWorkspace	GroupWorkspace
Container ID	91896bcb-162b-40dc-8f57-fc573ce30f50	6638bbe74ee49b57ef0a4702
Container Type	Wiki	TaskContainer
SocialDocument ID | SD-ID	a34fb7b1-619f-4d14-942b-65b33798cbd0	6638bbf19698217daeae9cde
IntellectualEntity ID | IE-ID	cd1543b5-de23-45c1-a894-f7da483240d8	6638bbf19698217daeae9cde
IntellectualEntity Name	How to paint mountains?	Paint Kilimanjaro on canvas
IntellectualEntity Type | IE-Type	WikiPage	Task
IntellectualComponent ID | IC-ID	ba1493b5-40de-45b5-b784-b7ed458246d8	Comment
IntellectualComponent Type | IC-Type	Attachment	663a2abd7f145af5223ba955
SimpleComponent ID | SC-ID	NULL	663a2ac082e5c1bf664c7ce2

SimpleComponent Type | SC-Type	NULL	Reaction
Event Action (CRUD) | E-Action	CREATED	CREATED
Event ID | E-ID	88190212	663a2ac04ee49b57ef0a7e98
Event Timestamp (local)	2024-06-17T13:21:04.829	2024-06-17T14:12:54.515
Event Timestamp (UTC)	2024-06-17T15:21:04.829	2024-06-17T16:12:54.515
Native Event Name	wiki.page.attachment.added	reaction.create
Activity (IE-Type.*IC-Type.SC-Type.****E-Action)***	WikiPage.Attachment. CREATED	Task.Comment.Reaction. CREATED

The idea of a log processor. As mentioned earlier, the transformation of the raw event data into a harmonised C-Log requires a tailored "transformation adaptor" for each included software product. Our first version of a *log processor* application supports the three phases recommended by DaProXSA (Just et al., 2024). We identified the available data sources in CNX, selected *relevant* data and then performed queries on the CNX databases. The relevance was determined by the C-Log attributes that are based on the *Collaborative Actions on Documents Ontology* (ColActDOnt). We adapted the schema (e.g. by adding local and UTC timestamps) and applied it to each event record to augment the native log (e.g. names of spaces, IDs of containers) to capture additional process-related information. Finally, the enriched event logs were exported into one harmonised multi-system data store for cross-system analyses using the *SPM Cockpit Alpha*.

SPM Cockpit Alpha. In parallel with the extensive data processing in this cycle, the ADR Researchers developed the alpha version of the SPM Cockpit. The app is meant as a specialised tool that imports event data of ECS and performs Process Mining techniques to create process models that represent collaborative activity in ECS. The Alpha version, which was developed and tested in this cycle was used primarily with the concept of the workspace as a CaseID. At the start of the phase, the events were on a low level of abstraction. Not surprisingly, this still did not lead to satisfactory results. However, the developed approach behind this tool was promising, so with the enriched data available at the end of cycle 2, we revised the tool into the beta version. This will be described in the following section.

4.3 Cycle 3: "Addressing the Description Challenge" – Collaborative Work Scenes

Project activity. In the Beta version, we added a *graphical user interface (GUI)* to facilitate access for the UCT Management Team for evaluations. The enhanced event data was imported into the SPM Cockpit and used to generate process models, which were then transformed into collaboration patterns. These patterns were discussed in multiple workshops between the UCT Management team and the ADR Researchers. The UCT team provided descriptions of typical work scenes carried out in UniConnect in natural language. The descriptions of these scenes were mapped to the patterns and added to an SPM Catalogue.

Research findings. The SPM Method aims to discover collaboration processes from the events recorded during the use of (collaboration-supporting) software systems. The basic concept of collaboration lies in *the joint work of two or more participants on a common goal* (Biuk-Aghai et al., 2005). The quality of the method is ultimately determined by its ability to accurately reflect

the human-triggered real-world collaboration processes. Thus, the final cycle aimed to provide *visualisations* and *descriptions* of work processes that can be used to discuss the conformance with the actions of people in the real world. For this, the SPM Cockpit was further developed to extract and visualise *collaboration patterns,* which can be mapped to work scenes that are familiar to people because they occur in their everyday work. This allows us to compare the *discovered scenes* (from SPM) with *real-world practices* in collaborative work (the observations of people).

What arises is the question of how to represent such collaboration related to collaborative work in ECS. As we have established above, collaboration is expressed through joint actions around social documents. A social document (in the form of a growing document graph e. g. a board post, comments to this board post or tags added to the board post) is the output artefact of a collaborative activity (e. g. discussing a topic). Thus, the idea is to construct the visualisation around each of the social documents, which we name the *collaboration process instance*. Each social document exists in a container (e. g. the forum), which, again, exists in a space (e. g. a project workspace). The events in the C-Log contain information about these relations, which means that we can construct traces of events based on the unique identification of the social document by the *SpaceID*, the *ContainerID* and the *SocDocID*. Using this composed identifier as a CaseID is an important aspect of our suggested approach and an innovation for process discovery in general, as this concept is not covered in any other research on the application of Process Mining with events derived from collaborative software. Another important aspect is the term *work*, which is expressed through the conducted activities on a social document. For this purpose, each event in the C-Log is assigned to a defined activity (e. g. “Blog Post Created”, “Blog Post Updated”, or “Blog Post Comment Created”). Thus, we can use this attribute to describe the *work* in our collaboration process instance. Finally, the terms “*joint*” and “*of two or more participants*” are relevant. This brings us to the idea of using the agent in the C-Log, as this event attribute defines the contributing participants. To do so, we combine the original activity with the agent and thereby construct a classifier in the form of “{Activity} by {Agent(s)}” for each event to express that a certain participant (the agent) executes a given collaborative activity. This means that the later visualisation of the collaboration process instance can convey the joint work of two or more participants. In terms of the Process Mining domain, we combine the *control-flow* and the *organisational perspective*. Note that we rename each agent based on the occurrence in the trace of the respective agent, i. e. we do not use their ID (or name), instead, we rename the initial agent to “Resource 1”, the second agent to “Resource 2”, etc. This step is required for the pattern detection approach in a later step.

We now have sequences of events grouped by the social document and labelled according to their collaborative work activity. Traditionally, such an event log serves as input for process discovery, i. e. state-of-the-art algorithms, e. g. the Heuristic Miner (Weijters et al., 2006) or the Inductive Miner (Leemans et al., 2013), process the sequences of such events into process models. These algorithms require event logs with a multi-set of traces to construct graph-based process models that show the underlying process semantic (van der Aalst, 2022a), for instance, in the form of a Petri Net or a BPMN diagram. However, because users in an ECS behave in an ad hoc manner, such process models provide a semantic that has no meaning for such a collaboration process because, for instance, there is no parallel semantics in collaborative work activities. Surely, we note that there exist synchronous collaboration tools (e. g. working with multiple authors on a Word document at the same time or conducting a video conference). However, we assume such

activities as single "joint" activities (e.g. "Word Document Updated by A and B"). Furthermore, due to the nature of the ad hoc behaviour, most traces are different in their appearance, i.e. there are too many possible trace variants, so traditional process discovery still produces spaghetti models, which do not provide value for collaborative work analysis. Based on the results of the previous cycles, we thus decided to follow a slightly different approach. We create for each trace, i.e. for each social document, a separate graph, as each trace represents a single instance of the collaboration. To visualise them, we straightforwardly mine a process map (van der Aalst, 2022a) for each of them, i.e., we construct a directed graph, where each event label (the activity) is converted to a node, such that each node represents the collaborative activity of a certain participant. Furthermore, based on the following relations of the events in such a trace, we add arcs connecting the nodes. In doing so, we construct directed graphs where we visualise the activities of participants and see the responding activities of other participants.

The discovery of the graph-based collaboration process instances is only an intermediate result. On the one hand, they show us concrete instances of how users actually work together in ECS, on the other hand, however, we want to make general claims about collaboration processes and practices. Thus, we use these graphs in a further step to detect *collaboration patterns*. Generally, a pattern "describes a problem which occurs over and over again in our environment, and then describes the core of the solution to that problem" (Alexander et al., 1977). Thus, the idea is to extract collaboration patterns in terms of a solution for collaborative problems (i.e., how to work collaboratively in order to reach a given common aim). Therefore, we apply *frequent subgraph mining (FSM)*, e.g. gSpan (Yan & Han, 2002) to detect common structures in the graph-based representations of the collaboration process instances. The resulting set of subgraphs can then be seen as a collection of collaboration patterns. FSM has promising results for pattern detection for similar process structures (Breuker et al., 2016; Diamantini et al., 2016). Furthermore, extending existing FSM algorithms with a relaxed component helps us to detect patterns that may not only frequently occur exactly the same way but are also similar in their form. This extends the resulting set of patterns so that interesting patterns are not missed.

Unfortunately, the presented approach leads to the discovery of many subgraphs, which in turn leads to the discovery of many similar patterns with similar meanings (similar collaboration scenes can be instantiated through similar structures of sequences of activities). Fortunately, this can be solved using state-of-the-art clustering algorithms. After the clustering step, we can give the clusters a name and a description based on the purpose/content of the related/containing patterns because each cluster has patterns with a similar structure (i.e. they are similar in their meaning). The resulting annotated clusters are collected and stored in a *Collaboration Pattern Repository*.

The Collaboration Pattern Repository can be shared as a collection of reference models for collaborative work activity as the clustered and annotated patterns provide solutions for collaboration problems. The patterns can now be used to quantify collaborative work activity by calculating metrics based on the patterns and the discovered collaboration process instances.

Mapping the collaboration patterns to the real world. In Section 2.2, we introduced the idea of a collaboration process, and now, with the help of the discovered collaboration patterns, we can describe the (previously unknown and abstract) collaboration scenes. Fig. 18 shows how the functional components of an ECS are assembled in group spaces to provide the necessary functionality for the digital support of collaborative work tasks. There, the use of the software is

conceptualised in work scenes. These scenes can be seen as single steps in the digital collaborative work and each (digital) scene can be mapped to a clustered set of (discovered) collaboration patterns. In this example, the work scene *discussing a topic* is expressed by the discovered collaboration pattern.

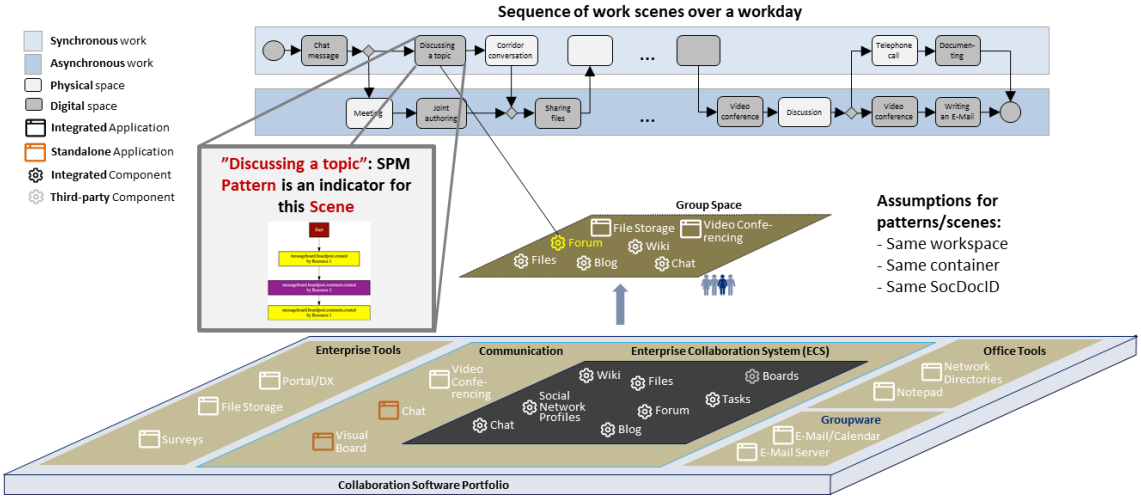
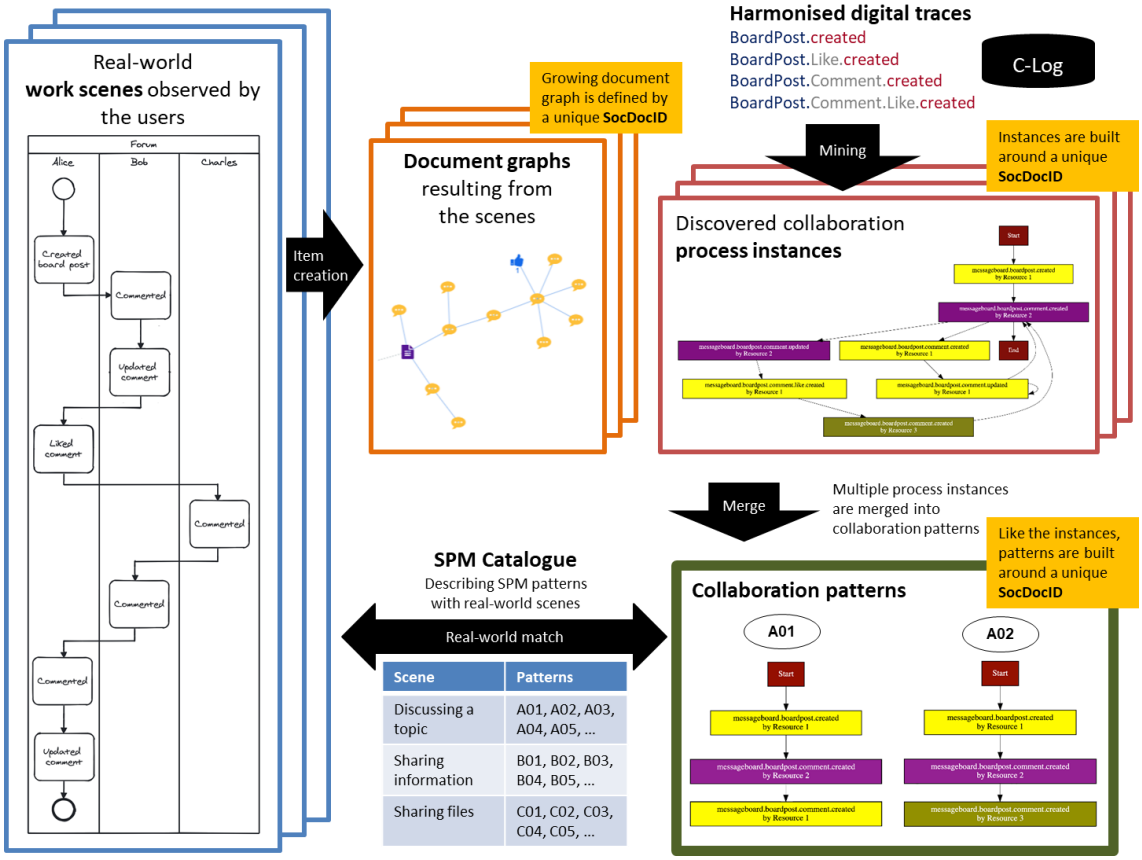


Fig. 18: Patterns are used to identify the work scenes that are supported by collaboration software

Resemblance between observed real-world processes and trace-based SPM models. In the last cycle of our project, we compared the descriptions of typical work scenes that are supported by UniConnect with the patterns discovered with the help of the SPM Cockpit. We were able to assign distinct descriptors to the scenes provided by the users in natural language. Fig. 19 shows an overview of the most important artefacts of our work. Users use collaboration software (collaboratively) in specific work scenes (e.g. for discussing a topic in the example).

In these scenes, the involved employees work jointly on documents and their actions on items create a (growing) document graph. The use of system functionality is recorded as digital traces in the event log. As described above, we transform (harmonise) the event records of the ECS into the C-Log schema using the DaProXSA approach. The C-Log events are then imported into the SPM Cockpit. The App discovers process instances and merges similar ones into patterns (A01, A02, A03, …). The patterns are compared to the real-world scenes (e.g. *discussing a topic*). Using the trace data from UniConnect, we found a high resemblance and were able to assign descriptive labels to the resulting patterns.

Development of an SPM Catalogue. The findings are collected in the *SPM Catalogue*. This catalogue (included in the SPM Cockpit) is a growing collaboration pattern repository to which we will be adding the patterns of other ECS apps in the future. The final objective is to include all functional areas of collaboration (shown in Fig. 1).



Scene	Patterns
Discussing a topic	A01, A02, A03, A04, A05, ...
Sharing information	B01, B02, B03, B04, B05, ...
Sharing files	C01, C02, C03, C04, C05, ...

Fig. 19: Overview of created ADR artefacts

5 Conclusions and Future Work

In this article, we address the two fundamental challenges that researchers encounter when they analyse the digital traces that are generated when people work with collaboration systems, the *data challenge* and the *description challenge*. Not surprisingly, these two challenges correspond to the bottom two levels ("Process Data" and "Discovery") of the Process Science Research Framework (vom Brocke et al., 2024). Our article is both *investigative* and *solution-oriented* in nature.

The findings from our ADR project showed that event data from collaboration software has to go through an intricate transformation and enhancement process before it can be used as research data in analytics. The examination of multiple leading ECS showed that the log formats of commonly used software products have very little in common and that we need to develop a *log pre-processor software* with a specialised *adaptor for every involved software product*. The application of existing Process Mining techniques is contingent on a viable CaseID. Experiments with multiple CaseID candidates showed that the SocDocID can be used to extract process models that reflect *actual* collaboration (work interactions between users). However, this ID is currently not contained in the native logs of software products and has to be artificially generated. Unfortunately, preparing the data and providing the SocDocID only takes us half way. When applying existing Process Mining techniques, the process instances of (supposedly identical) collaboration activity all look slightly different (or in PM terminology "all variants are unique"). This made it necessary to condense the single instances into patterns. With the help of the users who had carried out the processes, we were able to assign *descriptive labels* to these patterns and store the results in an extensible *SPM Catalogue*. SPM Cockpit[5] and SPM Catalogue have both been published for other researchers to use and build upon the current versions.

Our work has so far been limited to the analysis of a single integrated Enterprise Collaboration System (HCL Connections). As shown in Fig. 1 (portfolio of collaboration software), it is our long-term goal to be able to trace consecutive scenes of work over multiple involved systems which requires to include the full spectrum of functionality (and thus additional scenes) such as e-mail, chat, video conferencing, etc. For this, we are currently preparing the necessary adaptors for the log processor for data preparation and are building a harmonised central data store (C-Log Store). With the inclusion of further software products and additional functionality, our PM Catalogue will grow over time. In the long run, we hope to be able to compare the collaborative practices across organisational units and maybe find patterns of established practices. We will also continue to validate the derived patterns and their description by means of user interviews in our quest to describe real-world processes with the event data stored in the collaboration software portfolios of user organisations.

[5] w3id.org/spm/cockpit

6 References

Alberts, J., Blankenberg, C., & Williams, S. P. (2023). Identification and Classification of Adoption Supporting Measures for Enterprise Collaboration Systems. *Procedia Computer Science*, *219*, 319–329. https://doi.org/10.1016/j.procs.2023.01.296

Alexander, C., Ishikawa, S., & Silverstein, M. (1977). *A pattern language: towns, buildings, construction*. Oxford University Press.

Arazy, O., Lindberg, A., Lev, S., Wu, K., & Yarovoy, A. (2020). Emergent Routines in Peer-Production. *ACM Transactions on Social Computing*, *3*(1), 1–24. https://doi.org/10.1145/3366711

Baptista, J., Stein, M. K., Klein, S., Watson-Manheim, M. B., & Lee, J. (2020). Digital work and organisational transformation: Emergent Digital/Human work configurations in modern organisations. *Journal of Strategic Information Systems*, *29*(2). https://doi.org/10.1016/j.jsis.2020.101618

Berente, N., Seidel, S., & Safadi, H. (2019). Data-driven computationally intensive theory development. *Information Systems Research*, *30*(1), 50–64. https://doi.org/10.1287/isre.2018.0774

Biuk-Aghai, R. P., Simoff, S. J., & Debenham, J. (2005). From Ad-hoc to engineered collaboration in virtual workspaces. *11th Americas Conference on Information Systems (AMCIS 2005)*, 130–139.

Blatt, J., Delfmann, P., & Schubert, P. (2023). Event Abstraction for Enterprise Collaboration Systems to Support Social Process Mining. *ArXiv Preprint*, *2308.04396*, 1–8. http://arxiv.org/abs/2308.04396

Bose, R. P. J. C., Mans, R. S., & Van Der Aalst, W. M. P. (2013). Wanna Improve Process Mining Results? It's High Time We Consider Data Quality Issues Seriously. *IEEE Symposium on Computational Intelligence and Data Mining (CIDM)*, 127–134. https://doi.org/10.1109/CIDM.2013.6597227

Bose, R. P. J. C., & van der Aalst, W. M. P. (2009). Abstractions in process mining: A taxonomy of patterns. In U. Dayal, J. Eder, J. Koehler, & H. A. Reijers (Eds.), *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics): Vol. 5701 LNCS* (Issue June 2014, pp. 159–175). Springer. https://doi.org/10.1007/978-3-642-03848-8_12

Breuker, D., Matzner, M., Delfmann, P., & Becker, J. (2016). Comprehensible Predictive Models for Business Processes. *MIS Quarterly*, *40*(4), 1009–1034.

Budner, P., Wurm, B., Rosenkranz, C., & Mendling, J. (2022). Towards Routines Mining – Designing and Implementing the Argos Miner, a Design Science Artifact for Studying Routine Dynamics with Process Mining. *55th Hawaii International Conference on System Sciences*, 6462–6471.

Bullinger-Hoffmann, A., Koch, M., Möslein, K., & Richter, A. (2021). Computer Supported Cooperative Work - Revisited. *I-Com*, *20*(30), 215–228. https://doi.org/10.1109/2.291295

Chapman, P., Clinton, J., Kerber, R., Khabaza, T., Reinartz, T., Shearer, C., & Wirth, R. (2000). *CRISP-DM 1.0: Step-by-step data mining guide*. CRISP-DM Consortium.

de Reuver, M., Sørensen, C., & Basole, R. C. (2017). The digital platform: a research agenda. *Journal of Information Technology*, 1–12.

Diamantini, C., Genga, L., & Potena, D. (2016). Behavioral process mining for unstructured processes. *Journal of Intelligent Information Systems*, *47*(1), 5–32.

Diamantini, C., Genga, L., Potena, D., & Ribighini, G. (2014). A methodology for building log of collaboration processes. *2014 International Conference on Collaboration Technologies and Systems, CTS 2014*, 337–344. https://doi.org/10.1109/CTS.2014.6867586

Diba, K., Batoulis, K., Weidlich, M., & Weske, M. (2020). Extraction, correlation, and abstraction of event data for process mining. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, *10*(3), 1–24. https://doi.org/10.1002/widm.1346

Drodt, C., & Reuther, M. (2019). Predicting User Interaction in Enterprise Social Systems Using Process Mining. *Americas Conference on Information Systems (AMCIS)*, 1–10.

Fayyad, U., Piatetsky-shapiro, G., & Smyth, P. (1996). From Data Mining to Knowledge Discovery in Databases. *AI Magazine*, *17*(3), 37–54.

Ferreira, J., Claver, P., Pereira, P., & Thomaz, S. (2020). Remote Working and the Platform of the Future. In *Boston Consulting Group* (Issue October). https://pulse.microsoft.com/uploads/prod/2020/10/BCG-Remote-Working-and-the-Platform-of-the-Future-Oct-2020.pdf

Franzoi, S., & Grisold, T. (2023). Studying Dynamics and Change With Digital Trace Data: a Systematic Literature Review 1. *31st European Conference on Information Systems (ECIS 2023)*, 1–16.

Gerbl, J., & Williams, S. P. (2023). Identifying workgroup dimensions and their implications for the design of digital workspaces. *CENTERIS – International Conference on ENTERprise Information Systems*.

Greeven, C. S., & Williams, S. P. (2016). Enterprise Collaboration Systems: An Analysis and Classification of Adoption Challenges. *Procedia Computer Science*, *100*, 179–187. https://doi.org/10.1016/j.procs.2016.09.139

Günther, C. W., & Rozinat, A. (2012). Disco: Discover Your Processes. In N. Lohmann & S. Moser (Eds.), *Demonstration Track of the 10th International Conference on Business Process Management (BPM 2012)* (pp. 40–44).

Günther, C. W., & van der Aalst, W. M. P. (2007). Fuzzy Mining – Adaptive Process Simplification Based on Multi-perspective Metrics. *International Conference on Business Process Management*, 328–343.

Hacker, J., & Riemer, K. (2020). Identification of User Roles in Enterprise Social Networks: Method Development and Application. *Business and Information Systems Engineering*. https://doi.org/10.1007/s12599-020-00648-x

Hartl, S., Franzoi, S., Grisold, T., & Vom Brocke, J. (2023). Explaining Change with Digital Trace Data - A Framework for Temporal Bracketing. *56th Hawaii International Conference on System Sciences*, 5663–5672.

Janetzko, D. (2017). Nonreactive Data Collection Online (Chapter 5). In Nigel G. Fielding, R. M. Lee, & G. Blank (Eds.), *SAGE Handbook of Online Research Methods* (pp. 76–91).

Just, M., & Schubert, P. (2023). Collaborative Actions on Documents Ontology (ColActDOnt). *Procedia Computer Science*, *219*, 294–302. https://doi.org/10.1016/j.procs.2023.01.293

Just, M., Schubert, P., Blatt, J., & Delfmann, P. (2024). Data Preprocessing for Cross-System Analysis: The DaProXSA Approach. *Procedia Computer Science*, *239*, 1635–1644. https://doi.org/10.1016/j.procs.2024.06.340

Koceska, N., & Koceski, S. (2020). The importance of Enterprise Collaboration Systems during a pandemic. *Journal of Applied Economics and Business*, *8*(4), 35–41.

Leemans, S. J. J., Fahland, D., & van der Aalst, W. M. P. (2013). Discovering Block-Structured Process Models from Event Logs - A Constructive Approach. In J. Colom, JM., Desel (Ed.), *Application and Theory of Petri Nets and Concurrency. PETRI NETS 2013. Lecture Notes in Computer Science* (vol 7927). Springer.

Leonardi, P. M., Huysman, M., & Steinfield, C. (2013). Enterprise Social Media: Definition, History, and Prospects for the Study of Social Technologies in Organizations. *Journal of Computer-Mediated Communication*, *19*(1), 1–19. https://doi.org/10.1111/jcc4.12029

Mosen, J., Williams, S. P., & Schubert, P. (2020). Visualizing Social Documents as Traces of Collaborative Activity in Enterprise Collaboration Platforms. *53rd Hawaii International Conference on System Sciences*, 5369–5378.

Mosen, J., Williams, S. P., & Schubert, P. (2024). Work Practice Diversity in Enterprise Collaboration Systems: an Analysis of Social Documents. *Procedia Computer Science*, *239*, 1433–1440.

Nitschke, C. S., Hult, H. V., & Bigolin, F. (2020). Shared Workspaces of the Digital Workplace: From Design for Coordination to Coordination for Flexible Design. *53rd Hawaii International Conference on System Sciences*, 451–460. https://hdl.handle.net/10125/63795

Nitschke, C. S., & Williams, S. P. (2018). Traces of design activity: the design of coordination mechanisms in the shaping of enterprise collaboration systems. *Procedia Computer Science*, *138*, 580–586. https://doi.org/10.1016/j.procs.2018.10.078

Peffers, K., Tuunanen, T., & Niehaves, B. (2018). Design science research genres: introduction to the special issue on exemplars and criteria for applicable design science research. *European Journal of Information Systems*, *27*(2), 129–139. https://doi.org/10.1080/0960085X.2018.1458066

Pentland, B. T., Recker, J., Wolf, J., & Wyner, G. (2020). Bringing Context Inside Process Research with Digital Trace Data. *Journal of the Association for Information Systems*, *21*(5), 1214–1236.

Richter, A. (2020). Locked-down digital work. *International Journal of Information Management*, *55*. https://doi.org/10.1016/j.ijinfomgt.2020.102157

Richter, A., & Riemer, K. (2013). Malleable end-user software. *Business and Information Systems Engineering*, *5*(3), 195–197.

Riemer, K., Stieglitz, S., & Meske, C. (2015). From Top to Bottom: Investigating the Changing Role of Hierarchy in Enterprise Social Networks. *Business & Information Systems Engineering (BISE)*, *57*(3), 197–212.

Rozinat, A. (2013). *Disco Tour* (Issue May). http://fluxicon.com/disco/files/Disco-Tour.pdf

Schoch, M., Gimpel, H., Maier, A., & Neumeier, K. (2023). From broken habits to new intentions: how COVID-19 expands our knowledge on post-adoptive use behaviour of digital communication and collaboration. *European Journal of Information Systems*, *32*(6), 989–1010. https://doi.org/10.1080/0960085X.2022.2096489

Schön, H., Strahringer, S., Furrer, F., & Kühn, T. (2019). Business Role-Object Specification: A Language for Behavior-aware Structural Modeling of Business Objects. *Internationale Tagung Wirtschaftsinformatik*, 244–258. https://aisel.aisnet.org/wi2019/track03/papers/9/

Schubert, P. (2024a). Areas of Collaboration Analytics. *International Conference on ENTERprise Information Systems (CENTERIS 2024)*, 1–14.

Schubert, P. (2024b). IRECS Framework: Identification of Requirements for Enterprise Collaboration Systems. *Procedia Computer Science*, *239*, 1467–1475.

Schubert, P., & Williams, S. P. (2016). The Case of UniConnect - The Shaping of an Academic Collaboration Platform. *Multikonferenz Wirtschaftsinformatik*, 327–338.

Schubert, P., & Williams, S. P. (2022). Enterprise Collaboration Platform Configurations: an Empirical Study. *European Conference on Computer-Supported Cooperative Work*, 1–17.

Schubert, P., Williams, S. P., Just, M., Alberts, J. S., & Bahles, S. (2025). How Are Employees Using Collaboration Software to Support Their Work? A Method for Analyzing Digital Traces in Enterprise Collaboration Systems. *Hawaii International Conference on System Sciences*.

Schwade, F. (2021). Social Collaboration Analytics Framework: A framework for providing business intelligence on collaboration in the digital workplace. *Decision Support Systems*, *148*, 113587.

Schwade, F., & Richter, A. (2022). Forced Adoption: A new Phenomenon of Information Systems Adoption. *Americas Conference on Information Systems (AMCIS)*. https://aisel.aisnet.org/amcis2022/sig_cnow/sig_cnow/4

Schwade, F., & Schubert, P. (2017). Social Collaboration Analytics for Enterprise Collaboration Systems: Providing Business Intelligence on Collaboration Activities. *Hawaii International Conference on System Sciences*, 401–410.

Schwade, F., & Schubert, P. (2018a). A Survey on the Status Quo of Social Collaboration Analytics in Practice. *European Conference on Information Systems*, 1–15.

Schwade, F., & Schubert, P. (2018b). Social Collaboration Analytics for Enterprise Social Software: A Literature Review. *Multikonferenz Wirtschaftsinformatik 2018*, 1–12.

Schwade, F., & Schubert, P. (2019). Developing a User Typology for the Analysis of Participation in Enterprise Collaboration Systems. *Hawaii International Conference on System Sciences*, 460–469.

Sein, M. K., Henfridsson, O., Purao, S., Rossi, M., & Lindgren, R. (2011). Action design research. *MIS Quarterly: Management Information Systems*, *35*(1), 37–56. https://doi.org/10.2307/23043488

Simões, D., Antunes, P., Carriço, L., Simones, D., Antunes, P., & Carrico, L. (2018). Eliciting and Modeling Business Process Stories. *Business and Information Systems Engineering*, *60*(2), 115–132. https://doi.org/10.1007/S12599-017-0475-3

van der Aalst, W. M. P. (2005). Process mining in CSCW systems. *Proceedings of the 9th International Conference on Computer Supported Cooperative Work in Design*, *1*, 1–8. https://doi.org/10.1109/cscwd.2005.194134

van der Aalst, W. M. P. (2011). Process Mining. In *Process Mining*. Springer.

van der Aalst, W. M. P. (2016). *Process Mining: Data Science in Action*. Springer.

van der Aalst, W. M. P. (2018). Process discovery from event data: Relating models and logs through abstractions. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, *8*(3), 1–21. https://doi.org/10.1002/widm.1244

van der Aalst, W. M. P. (2022a). Foundations of Process Discovery. In W. M. P. van der Aalst & J. Carmona (Eds.), *Process Mining Handbook* (1st ed., pp. 37–75). Springer Cham. https://doi.org/10.1007/978-3-031-08848-3_2

van der Aalst, W. M. P. (2022b). Process Mining: A 360 Degree Overview. In W. M. P. van der Aalst & J. Carmona (Eds.), *Process Mining Handbook* (1st ed., pp. 3–34). Springer Cham. https://doi.org/10.1007/978-3-031-08848-3_1

van der Aalst, W. M. P., Adriansyah, A., De Medeiros, A. K. A., Arcieri, F., Baier, T., Blickle, T., Bose, J. C., Van Den Brand, P., Brandtjen, R., Buijs, J., Burattin, A., Carmona, J., Castellanos, M., Claes, J., Cook, J., Costantini, N., Curbera, F., Damiani, E., De Leoni, M., … Wynn, M. (2012). Process mining manifesto. *Lecture Notes in Business Information Processing*, *99 LNBIP*(PART 1), 169–194. https://doi.org/10.1007/978-3-642-28108-2_19

van Zelst, S. J., Mannhardt, F., de Leoni, M., & Koschmider, A. (2021). Event abstraction in process mining: literature review and taxonomy. *Granular Computing*, *6*, 719–736. https://doi.org/10.1007/s41066-020-00226-2

Vianna, D., Kalokyri, V., Borgida, A., Marian, A., & Nguyen, T. (2019). Searching heterogeneous personal digital traces. *Proceedings of the Association for Information Science and Technology*, *56*(1), 276–285. https://onlinelibrary.wiley.com/doi/10.1002/pra2.22

vom Brocke, J., Jans, M., Mendling, J., & Reijers, H. A. (2021). A Five-Level Framework for Research on Process Mining. *Business & Information Systems Engineering*, *63*(5), 483–490. https://doi.org/10.1007/s12599-021-00718-8

vom Brocke, J., van der Aalst, W. M., Grisold, T., Kremser, W., Mendling, J., Pentland, B., Recker, J., Roeglinger, M., Rosemann, M., & Weber, B. (2024). Process Science: The Interdisciplinary Study of Continuous Change. *Process Science*, *2*(1), 1–14. https://doi.org/https://doi.org/10.1007/s44311-024-00001-5

Weijters, A. J. M. M., van der Aalst, W. M. P., & de Medeiros;, A. K. A. (2006). Process Mining with the HeuristicsMiner Algorithm. *Beta Working Papers (TU Eindhoven)*, *166*(2006).

Weske, M. (2012). Business Process Management: Concepts, Languages, Architectures. In *Business Process Management* (2nd ed., Vol. 54). Springer. https://doi.org/10.1007/978-3-540-73522-9

Williams, S. P., & Grams, S. (2022). Remote Working Study 2022. *CEIR Research Report, No. 01/2022*, 1–24.

Williams, S. P., Mosen, J., & Schubert, P. (2020). The Structure of Social Documents. *53rd Hawaii International Conference on System Sciences*.

Williams, S. P., & Schubert, P. (2018). Designs for the Digital Workplace. *Procedia Computer Science*, *138*, 478–485.

Yan, X., & Han, J. (2002). gSpan: graph-based substructure pattern mining. *2002 IEEE International Conference on Data Mining*, 721–724. https://doi.org/10.1109/ICDM.2002.1184038

Zerbato, F., Seiger, R., Di Federico, G., Burattin, A., Weber, B., Federico, G. Di, Burattin, A., Di Federico, G., Burattin, A., Weber, B., Federico, G. Di, Burattin, A., Di Federico, G., Burattin, A., & Weber, B. (2021). Granularity in Process Mining: Can we fix it? *CEUR Workshop Proceedings*, *September*, 40–44.

The CEIR team and IndustryConnect

The Center for Enterprise Information Research (CEIR) at the University of Koblenz is a cooperation project between the Business Application Systems Research Group lead by Professor Dr Petra Schubert and the Enterprise Information Management Research Group lead by Professor Dr Susan P. Williams. CEIR has the aim of bringing together Industry and University in joint research projects, which are directed towards developing new theoretical insights as well as relevant findings that can be applied success-fully in practice.

The IndustryConnect initiative (industryconnect.de) was launched in 2015 by CEIR and facilitates the exchange of experiences among user companies under the moderation of the participating CEIR team members. IndustryConnect addresses current problems and issues in the area of collaborative work in companies using Enterprise Collaboration Systems. IndustryConnect goes beyond the usual experience groups, round tables or business lunches. The participating researchers continue their work on the topics between the meetings and make the results available in the form of documents, methods, techniques and guidelines.

Susan Williams
Professor
CEIR Research Director

Enterprise Information Management

Petra Schubert
Professor
CEIR Research Director

Business Application Systems

Sebastian Bahles
CEIR Researcher
Manager UCT & IndustryConnect

Workspace Management

Julian Mosen
CEIR Researcher
Head of IT

Social Documents

Simon Meier
CEIR Researcher

Ontology-based data access

Jens Alberts
CEIR Researcher

Work routines & coordinative practices

Martin Just
CEIR Researcher

Social Process Mining and Cross-system analysis

Jennifer Gerbl
CEIR Researcher

Hybrid Work and Digital Ethnography

Cornelia Mc Stay
Administrative assistant

CEIR REPORT

No. 02/2024

Social Process Mining: Deriving Collaborative Work Processes from the Event Data of Enterprise Collaboration Systems