Proceedings of the 53rd Hawaii International Conference on System Sciences | 2020

Analysis of Task Management in Virtual Academic Teams

Sabine Nagel
University of Koblenz-Landau
Institute for Information Systems Research
Germany
snagel@uni-koblenz.de

Florian Schwade
University of Koblenz-Landau
Institute for Information Systems Research
Germany
fschwade@uni-koblenz.de

Abstract

Social Collaboration Analytics (SCA) aims at measuring collaboration in Enterprise Collaboration Systems (ECS). In this paper, we apply SCA to investigate the use of Task Management (TM) features in virtual academic teams on a collaboration platform. This paper contributes to theory by developing the TM Catalog describing the elements and characteristics of TM. Our literature review identified only three studies analyzing the use of TM features in ECS. These studies base their analyses on transactional data (event logs). We propose to analyze both the structure and characteristics of tasks, as well as how tasks are used. In our paper, we show how SCA can be applied to gain insights on the use of TM features. Based on data from an academic collaboration platform, we demonstrate the characteristics of tasks and how different types of virtual academic teams make use of TM features.

1. Introduction

Understanding how individuals collaborate in groups has been of significant interest to computer-supported cooperative work (CSCW) scholars ever since [37]. The emergence of Social Media and the rise of Enterprise Collaboration Systems (ECS) has renewed interest in understanding how people collaborate using computer-mediated technologies in the digital workplace. Due to the increasing use of Social Media in private life, employees expect to use *socially-enabled* tools in their workplace [45]. Consequently, in recent years, we have witnessed the emergence of socially-enabled collaboration software such as HCL Connections, Microsoft SharePoint, or Atlassian Confluence [21]. ECS combine social features (e.g., social profiles, wikis, blogs, forums) with traditional groupware functionalities [39]. They support all areas of the *8C model for Enterprise Information Management*, which describes the main areas of collaboration [46], especially the inner core consisting of *communication*, *cooperation*, *combination,* and *coordination*. Therefore, companies introduce ECS for supporting communication and collaboration between employees [45]. Accordingly, at universities, academics and students can use such platforms for coordinating their lectures, research projects, and team works. Moreover, collaboration platforms support computer-supported collaborative learning (CSCL) [13] in academic institutions.

In this research, we investigate the aspect of *coordination* on the academic collaboration platform UniConnect by focusing on the use of Task Management (TM) features. As every project involves tasks, people, and deadlines, the success of a project heavily depends on TM [25]. Examples for this include appropriately prioritizing tasks, as well as time management [6, 25, 26]. Improper TM can also lead to an uncontrollable amount of unfinished tasks [7] and eventually, to a bad project outcome [25]. Bellotti et al. [1] argue that effective TM requires resources and knowledge on the use of such features.

Schubert and Glitsch [20] have identified typical use cases for analyzing and describing collaboration in companies. 8 out of 13 use cases involve coordination and TM-related actions, which emphasizes the importance of TM in the area of CSCW and ECS. Examples for such use cases include project or team organization, and software development.

In our research, we seek to demonstrate how Social Collaboration Analytics (SCA) can be applied to provide an understanding of TM usage in collaboration platforms. SCA is a systematic approach for automatically "analyzing and displaying collaboration activity of users in socially-enabled collaboration systems" [39:402] using database queries and other computational methods.

Based on the analysis of an academic collaboration platform, we (1) discuss the structure and characteristics of tasks and (2) demonstrate how different types of teams make use of TM features for coordination. Therefore, we define the following research questions:

1. What are the characteristics of TM, and how do modern ECS implement TM functionalities?

URI: https://hdl.handle.net/10125/63796
978-0-9981331-3-3

2. How can SCA be applied for measuring coordination activities in different types of (virtual) academic teams in ECS?
3. How do different types of (virtual) academic teams make use of coordination features in ECS?

RQ1 seeks to establish the terminology, characteristics, and components of TM based on a literature review. The outcome is a TM catalog describing the different types of content and actions related to TM. By addressing RQ2, we demonstrate how the methods and structures from SCA can be applied for understanding how different types of virtual teams make use of TM features. Finally, based on an analysis of data from an academic collaboration platform with more than 3500 users and more than 40 member institutions, we demonstrate which insights can be gained on the use of coordination features in digital collaboration platforms. This includes the executed actions, as well as the characteristics and structure of the content.

This paper is structured as follows: First, we introduce related work in the areas of TM and SCA (section 2). In section 3 we describe the research design of our work, followed by a condensed overview of the TM catalog (section 4). We then analyze TM-related content and actions using data from an existing ECS. We conclude with a discussion of the results and potential limitations, as well as an outlook into future research.

2. Terminology & Related Work

According to Moran [32:4], people tend to organize their work life into individual tasks or activities, which are defined as "a set of (mental or physical) actions carried out by people" to accomplish particular objectives. Geyer et al. [18:713] define an activity as a "logical unit of work that incorporates all the tools, people, and resources needed to get a job done".

In academic literature, the terminology in the area of TM differs widely. Even within single publications, authors use terms inconsistently. While the Term *Task Management* is used dominantly (in 90% of all sources that specifically name that concept), another common term is *Activity Management* (28%). The same applies to the terms *task* (95%) and *activity* (69%); however, authors also frequently use the term *to-do* (20%) in order to describe a task. The above-described terms are often used synonymously; however, in this research we will consistently use the terms *Task Management (TM)* and *task* when talking about these general concepts.

In order to conduct TM, people can make use of so-called *Task Management Systems (TMS)*. Authors distinguish between Personal Task Management (PTM) [22] and Collaborative Task Management (CTM) [28]. In this research, we will focus on *CTM*; specifically, on the TM module of an academic collaboration system.

We apply SCA to provide insights into how virtual teams use the available TM features. SCA makes use of (1) transactional, (2) content and (3) organizational data. *Transactional data* (e.g., log files) are the primary data source for SCA and contain all user actions, including the action type, information on the modified content, the user, and a timestamp. Besides the user-generated content, *content data* stores metadata and structural information about the content (e.g., parent-child relationships). *Organizational data* covers the structure and users of an organization [39].

A previous literature review on studies in the field of SCA [40] resulted in 85 publications, of which only 62 publications apply analytics to a data set. The literature review identified seven key themes for SCA: (1) measurement of system usage, (2) analysis of communities, (3) identification of types of users, (4) identification of expertise, (5) identification of usage patterns, (6) analysis of networks and (7) measuring organizational and cultural impacts of ECS. Our analyses will be guided by the themes measurement of system usage and analysis of communities.

Only 3 of the identified studies investigate the *use of TM features* in ECS [15, 16, 35]. The authors apply process mining algorithms for *extracting generalized collaboration patterns* between employees [15, 16] and a *generalized life cycle* of tasks [35] from transactional data. While transactional data is appropriate for identifying patterns and lifecycles, it only provides limited information on the status or structure of a task.

Among the 62 studies, only 5 studies combine transactional and content data. We argue that a holistic analysis of the use of TM features in ECS requires combining transactional and content data for analyzing the characteristics and structure of a task as well as how users work with tasks.

3. Research Design

The application of SCA to analyze and understand TM-related collaboration requires a theoretical foundation. This includes an understanding of the research field, as well as a blueprint, the analysis can follow. To prepare the necessary foundation and execute our analyses, we organized our research design in three phases.

The (1) *preparation* phase consists of two structured literature reviews. The literature review on SCA provided the SCA key themes and a classification of the studies according to the type of data used for analysis (see section 2). An additional exhaustive literature review following [4, 5, 48] established the current state of research on TM. We started with an initial keyword search using the commonly used term "Task Management" in the three databases ScienceDirect, ACM Digital Library and IEEE Explore. We deliberately picked such a general term in order to avoid unwanted restrictions of the resulting literature. After the first iteration, we also added the term "Activity Management" to the search keywords as the search results indicated that it is used as a common synonym.

The literature review provided the foundation for the (2) *development of the TM Catalog*. The keyword search was followed by iterating through a forward and backward snowballing until saturation was reached. The search process resulted in 64 publications. In parallel, we applied a multi-level coding approach [30] for extracting TM-related functionalities. In two iterations of initial coding, we identified and defined an appropriate and consistent structure of codes. The actual coding process consisted of two cycles. The first coding cycle was primarily based on *in vivo coding* in order to maintain the terminology used by the original authors and to avoid distortion due to the subjectivity of the coder. In the second coding cycle, the codes were then standardized based on the identified terminology, as described in section 2. Also, the codes were grouped in order to summarize and define TM-related features resulting in the *TM Catalog* (section 4).

The investigation of the actual use of TM features on UniConnect, an academic collaboration platform, is the main phase of this research. The analysis was guided by CRISP-DM [8]. The *business understanding* (section 5.1) provided the context for the use and context of UniConnect. In the *data understanding phase* (section 5.2), we established an understanding of the underlying databases and their structure. During the *data preparation* (section 5.2), we selected and filtered the relevant data. Based on the prepared data and the TM catalog, we prepared the scripts for the data *analysis* and executed the analyses, which were described and interpreted in the *evaluation* (sections 5.3 and 5.4).

4. Task Management Catalog

The primary purpose of the TM Catalog is to serve as a blueprint for the evaluation of TMS regarding their functionalities, as well as to form a basis for structured SCA. Currently, the catalog is still at an early stage as it is fully literature-based and has only been applied once to evaluate a system. Despite this early stage, we tried to ensure a certain level of validity by only including features that were mentioned by at least two independent authors. Additionally, we included minor revisions after the first application of the catalog.

The TM functionalities in this catalog are divided into four main themes. The first theme (*Objects*) contains the different core elements and components of a TMS, as well as their attributes. The properties and context of these objects are then further specified in the second theme (*Task Properties & Context*). The third theme (*Information & Representation*) deals with the communication and visualization of information about objects and related actions, including notifications and awareness functionalities, as well as views and visualizations. The last theme (*System Properties*) contains all functionalities that refer to the system itself as opposed to its content. Examples include system integration with other applications, as well as possibilities of personalization.

As the focus of this paper is the analysis of TM-related content and transactional data using SCA, the following sections focus on discussing the themes *Objects* and *Task Properties & Context* as the foundation of this research. The tables in this chapter contain a condensed version of the catalog and include only the content and actions applicable to UniConnect. Also, a reference count and selected references are displayed.

In general, Objects can be divided into three subcategories: Core Elements, Components and Attributes.

The core element (Table 1) with the highest level of abstraction is a *project*. Projects usually comprise other TM-related objects, the central objects of a TMS being so-called *tasks*. Inside a project, *tasks* can be organized in *groups* or *lists*. In addition to *tasks*, TMS also often support textual items, which are referred to as *notes*. *Templates* facilitate reusing tasks.

Table 1: Core elements

Element	Ref. Count	Ref.
Project	8	[9, 12, 25]
Task	64	[11, 18, 32]
Group/List	27	[1, 22, 27, 28, 34]
Note	11	[1, 11, 41]
Template	19	[11, 18, 23]

In order to include further information to an element, additional components such as *attachments*, *tags* and *comments* can be added (Table 2).

Table 2: Components

Element	Ref. Count	Ref.
Attachment	24	[11, 24, 28]
Tag	9	[10, 43, 44]
Comment	11	[25, 27, 43]

Both core elements and components have attributes that further specify the respective object (Table 3). While authors suggest a variety of potential attributes, the most common ones include a *title*, a *description*, and a *deadline*. Additionally, one or more *members* can take on different *roles*, and *priorities* can be set to represent the importance of a task.

Table 3: Attributes

Element	Ref. Count	Ref.
Title	15	[11, 25, 28]
Description	23	[11, 14, 23, 27]
Deadline	32	[3, 24, 25, 28, 32]
Members & Roles	23	[11, 14, 18, 23, 28, 31]
Priority	27	[1, 17, 22, 26]

In addition to their attributes, TM-related objects can be characterized further by specifying their properties and context (Table 4). A task can either be of personal or collaborative nature (*type*). Among the most cited properties are *relationships* between tasks. This includes their position in a hierarchy, especially as tasks are often further decomposed into subtasks, and dependencies between two or more tasks. Furthermore, the *access* or *visibility* of tasks is often restricted based on an authorization model. Another important property is the *state* of a task. This includes the completion state (complete vs. incomplete) but also to the fact whether a task is assigned to another user or unassigned.

Table 4: Properties

Element	Ref. Count	Ref.
Type	8	[28, 36]
Relationships	33	[14, 23, 32, 36]
Access/Visibility	12	[11, 34]
State	20	[1, 28, 32, 36]

In addition to content, the TM catalog also contains actions that, in combination with the objects described above, represent transactions in a TMS.

Table 5: Task Management actions

	TM Actions	Ref. Count	Ref.
C	create (E) add (C, A) set (A, P) define (P)	45	[1, 9, 10, 11, 18, 19, 23, 25, 28]
	decompose (E)	20	[12, 25, 29, 33]
	copy (E, C)	4	[24, 28]
R	follow/unfollow (E)	2	[11, 18]
	search/filter/sort (E)	20	[3, 18, 25, 28]
U	edit (E, C, P) change (A, P)	27	[1, 11, 28, 34]
	assign/unassign (E)	24	[1, 23, 27, 42]
	(un-)complete (E)	28	[11, 19, 27, 42]
	move (E, C)	9	[2, 22, 28]
D	delete (E, C)	12	[11, 18, 23]

Table 5 contains a mapping of the identified TM actions to the corresponding CRUD (create, read, update, delete) actions, which describe basic data operations. Also, potential subjects of each action, namely element (E), component (C), attribute (A) and property (P), are specified, and selected references are listed.

There are different possibilities of creating content in a TMS. While elements always have to be *created* from scratch, other content such as components or attributes can be *added* to existing elements. Furthermore, users can *set* attributes (e.g., deadline) or *define* properties (e.g., relationships) of an element. Further actions that have been categorized as a creation of content include decomposing elements, as well as copying existing content. The catalog does not explicitly consider the consumption of content. However, actions such as *following* content (e.g., in order to be notified about changes) as well as *searching*, *filtering*, or *sorting* are closely related to consumption.

We differentiate between *editing* or *changing* content itself, changing the location of content, and updating the status of an element. In the latter case, elements can either be assigned to one or more users (change of assignment state) or completed (change of completion state). Finally, there is also the possibility of *deleting* elements or components from the system.

5. Social Collaboration Analytics

The following sections describe the analysis, which was guided by the phases of the CRISP-DM [8] as described in section 3.

5.1 Business Understanding

UniConnect is an academic collaboration platform hosted by the University Competence Center for Collaboration Technologies (UCT) [38]. The platform is based on HCL Connections (formerly: IBM Connections) and provides “opportunities for universities to work together and with industry partners on joint projects” [38]. The target user groups of UniConnect are students and academics, as well as their industry partners. Currently, UniConnect has more than 3500 users who represent more than 40 different member institutions from all over Europe.

Communities serve as joint workspaces for groups and thus represent teams. We identified different types of communities, which are used for different activities:

Class organization: The primary purpose of these communities is the organization of classes throughout a semester. Therefore, the involved users include all students taking the respective class, as well as the responsible academic staff. The main contents of communities are lecture materials provided by the academic staff. Students can download the materials and discuss the course content in forums.

Student projects: Many classes require students to form groups and collaborate on projects throughout the semester. In this case, communities support students in organizing and collaborating on these projects. Especially distributed teams (e.g. students of distance learning universities) benefit from these communities as the community serves as their central collaboration space.

Thesis organization: Most study programs are concluded by writing a thesis. This work is also often organized in communities that contain the student writing the thesis as well as the supervisors. Here, the student can use the features of UniConnect for self-organization but can also benefit from the collaborative aspects of the platform by involving the supervisors in their work, e.g. by discussing questions, providing regular updates and receiving feedback.

Ongoing academic research: In contrast to the previous communities, which have a defined lifespan (e.g. the length of a semester), ongoing academic research communities are permanent and consist of dispersed project partners. Research groups use communities for organizing their research projects and activities and for managing long-term projects. A prime example is the university-industry initiative IndustryConnect, which is led by a group of researchers and consists of more than 30 ECS user companies. In between the bi-annual workshops, the online community is used for documenting results and fostering the mutual exchange between the participants [47].

Testing: In addition to serving as a platform to facilitate collaboration, UniConnect is also a subject to research itself. Therefore, many test communities exist in which academics and students experiment with the system's features. As testing communities are not used for collaboration, they must be excluded from analyses.

5.2 Data Understanding & Data Preparation

As an integrated ECS, UniConnect consists of different modules, including Blogs, Wikis and a TM module (called Activities), which is the focus of this work (for a full list and description of all available modules see [39]).

While the majority of terminology regarding TM functionalities in UniConnect aligns with the concepts introduced in the catalog (section 4), the terminology regarding core elements differs slightly. Therefore, they will be mapped to the elements introduced in the TM catalog as well as briefly described regarding their functionality. In the TM module inside of communities, so-called Activities can be added. Activities describe high-level *tasks* that are mainly used to decompose a project into work packages. Each Activity can then be further decomposed into To Do Items (*tasks*) and Entries (textual items). While To Do Items and Entries share a majority of their characteristics, assignees and due dates can only be added to To Do Items. Sections are used to *group* To Do Items and Entries.

As each module of UniConnect has its own database, the following analyses will be based on the database of the TM module, namely the OPNACT database. Additionally, we will use the respective records of the METRICS database, which contains the transactional data for all modules.

Figure 1 shows an excerpt of the most important tables from the databases and demonstrates how these can be joined to enrich transactional data with content data and vice versa. The OPNACT database is organized in a *hierarchical* node structure. All Activities, To Do Items, Entries, Sections and Comments are stored as nodes in the OA_NODE table, including their metadata. The OA_TREE table defines the hierarchical relationship between the nodes. The OA_TAG table associates tags with nodes and the OA_ASSIGNEE table contains information on the assignee of a task.
In contrast, the METRICS database is organized in a *star* scheme, as visualized in Figure 1. Here, the F_TRX_EVENTS table is the central fact table containing the user actions and consists of condensed information about users, the item types and the type of event. Most information is stored in the form of IDs. The surrounding dimension tables map these IDs to human-readable information.

As shown in Figure 1, the *nodeuuid* from the OPNACT database and the *item_uuid* from the METRICS database can be used to join transactional and content data. Additionally, the user ids could be used for joining the data.

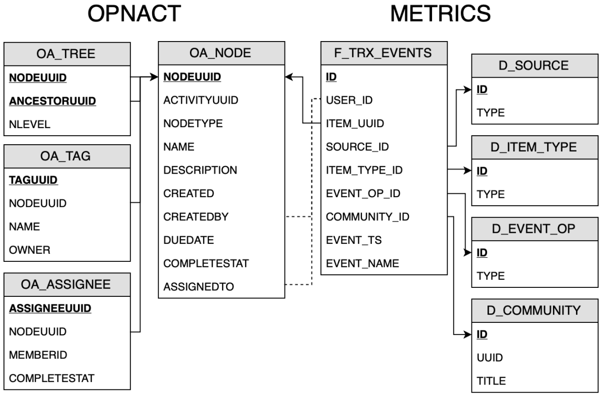

Figure 1: Connection between OPNACT and METRICS databases

The following vignette provides an example of how the content and transactional databases work.

Luke and Jack use the community 'Social Collaboration Analytics' for organizing their research. After *their last meeting, Luke creates the To Do Item 'Write project report' and assigns it to Jack. He defines the 4th of May 2019 as the deadline.*

Upon creating the To Do Item, a new node with the *nodetype* 'task', a unique *nodeuuid* and the *name* 'Write project report' is created and saved to the OPNACT database. Additionally, the task is associated with its ancestor Activity (*activityuuid*). Moreover, the content database contains information on the date of creation (*created*), the creator (*createdby*: Luke), the assignee (*assignedto*: Jack) and the defined *duedate* (4th of May 2019). On creation, the status (*completestat*) of the task is 'incomplete'. Simultaneously, the F_TRX_EVENTS table in the METRICS database logs a new event record with the relevant transactional data. As Luke created a new To Do Item, the record contains his *user_id*, the id of the created To Do Item (*itemuuid*) and the timestamp (*event_ts*) of creation. Further columns include the *source_id* which describes the module the event occurred in (Activities), the type of item that was created (*item_type_id*: task), the event type (*event_op_id*: create) and the id of the community in which the task has been created (*community_id*). The *event_name*, 'activities.task.created' is composed of the source, the item type and the event type.

The dimensions for SCA [39] guide the development of questions and queries for SCA as follows:

- *Level of analysis*: In order to get an overview of the entire system and its content, we will start with analyses on the platform level. As collaboration happens mostly within communities, we will then look at the different community types by analyzing transactional data.
- *Content type*: As the focus of this work are TM-related content and actions, we will limit our analyses to the Activities module of UniConnect.
- *Content components*: We will look at all objects (i.e. node types) that can occur in the context of TM. The different types of nodes will be further specified in section 5.3.
- *Action type*: In addition to all TM-related content, we will also consider potential event types, which will be described in more detail in section 5.4.
- *Time and further filters*: As UniConnect is an academic research platform, most of its collaboration follows a semester structure. Therefore, we will limit transactional data to a timeframe of three semesters, from 04/01/17 until 09/30/18. Additionally, we will make use of the filters described below.

As already mentioned in the previous section, UniConnect and therefore, its databases contain test data in addition to data resulting from actual collaboration activities. In order to get accurate results when conducting SCA, we excluded test communities, as well as test data outside of communities. Apart from omitting test data, no other filters such as limitations to specific user groups or communities were applied.

Table 6 gives an overview of the remaining data, which will be used as a basis for the analyses in the next chapter. After applying the filter described above, there were 9207 nodes left in the OA_NODE table. While this database only contains data from the Activities module, the events in the F_TRX_EVENTS table cover all modules. From a total number of 1,209,277 events across the entire system, 1.4% of them (16,988) occurred in the Activities module. The number of events in a module does not necessarily represent its actual usage, as events are created differently in each module, depending on its nature and implementation.

Table 6: Overview of underlying data

	System	Activities	Ratio
Nodes	9,207	9,207	100%
Events	1,209,277	16,988	1.4%

The following sections describe the analysis results for content and transactional data in the context of TM.

5.3 Analysis of System Content (content)

Table 7 provides an overview of the distribution of different types of nodes from the OA_NODES table.

Table 7: Distribution of node types

Type	%
Activities in communities	7.29%
Activities outside of communities	3.61%
Sections	13.92%
To Do Items	49.43%
Entries	10.32%
Replies (Comments)	11.05%
Fields[1] (Attachments)	3.20%
Membership, Templates, Links to Activities	1,18%

The numbers show that To Do Items (49.43%) are the central element in the Activities module. Interestingly, the share of Activities in communities (7.29%) is higher than the share of Activities outside of communities (3.61%). As Activities outside of communities are mostly used for self-organization, the results demonstrate that users prefer to use Activities for CTM. Sections, Entries and Replies occur with similar frequency.

Next, we examine the average characteristics of the node types Activities, To Do Items and Entries (Table 8). 67.58% of Entries have a description, which only

[1] In the UniConnect database, all fields are of type *attachment*. However, other possible types include *date*, *person*, *link to file/folder*, *bookmark* or icon.

applies to 54.44% of Activities and 46.23% of To Do Items. This shows that descriptions are less important on the level of To Do Items, as their titles should already represent actionable steps. The average length of descriptions is 57 characters for Activities, 387 characters for To Do Items and 542 characters for Entries. This is not surprising as Entries represent textual items. Although less To Do Items than Activities contain a description, the description of To Do Items is longer.

In UniConnect, Activities and To Do Items are the only elements that can have a due date. Depending on the context, the system forces the user to set a due date for a new Activity. Thus, the share of Activities having a due date does not allow further interpretation. Interestingly, only 52.78% of the To Do Items have a due date. This is surprising as the TM Catalog identified due dates as essential for successful TM.

After inactivity for more than three months, the system automatically completes Activities. As 24% of all completed Activities have been autocompleted, this explains the high share of completed Activities (84.65%). However, To Do Items are not automatically completed on the completion of the parent Activity.

On the average, an Activity is completed 16.5 days after its creation. To Do Items are completed after 18.4 days on the average. However, the median is significantly lower for both durations, being at 3.8 days for Activities and 2.5 days for To Do Items.

57.97% of all To Do Items are assigned to one or more users, the average being at 1.3 assignees.

Table 8: Average attributes and properties of Activities, To Do Items & Entries

	Activity	To Do Item	Entry
Description	54.44%	46.23%	67.58%
Due Date	70.89%	52.78%	-
Completion Status	84.65%	54.87%	-
Assignment Status	-	57.97%	-

The next section provides insights on the structure of Activities, To Do Items and Entries by investigating their components (Table 9).

Only 1.91% of To Do Items and 8.53% of Entries have attachments (files or links). The higher number of Entries with attachments can again be traced back to the textual nature of Entries.

While 42.27% of all Activities are tagged, this number is significantly lower for To Do Items (14.85%) and Entries (21.37%). Activities, To Do Items and Entries each have 3 tags on average. Activities and Entries both show a mode of 1 tag. The mode for tags added to To Do Items is 3. The median for tags of Activities and Entries is 2, and 3 for To Do Items.

11.23% of To Do Items and 8.11% of Entries contain comments. However, commented elements usually only have one comment (average, mode and median: 1). Therefore, our observations show that comments are preferably used to annotate or give a quick update about a task than to have an actual discussion.

Table 9: Average components of Activities, To Do Items & Entries

	Activity	To Do Item	Entry
Attachment	-	1.91%	8.53%
Tags	42.27%	14.85%	21.37%
Comment	-	11.23%	8.11%

Next, we investigate the hierarchy of Activities and To Do Items (*decomposition*). On average, an Activity has a depth of 1.8, i.e., about two levels of sub-elements. In contrast, the average depth of To Do Items is 0.23. This shows that To Do Items usually do not have further subtasks. On average, an Activity has 9 (mode: 4) sub-elements. To Do Items only have 0.36 sub-elements (mode: 0). This demonstrates that Activities are high-level tasks that are decomposed into subtasks. To Do Items usually do not require any further decomposition as these represent actionable steps.

5.4 Analysis of System Usage (transactional)

Figure 2 shows the distribution of the 8 most frequent events that can occur in the Activities module.

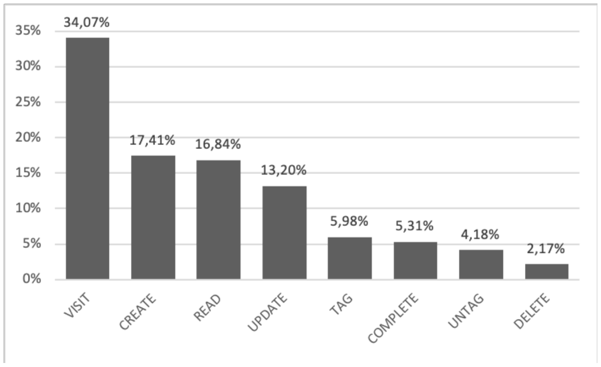

Figure 2: Event distribution in the Activities module

With 34.07%, VISIT events are the most frequently occurring event. This is followed by CREATE (17.41%), READ (16.84%) and UPDATE (13.2%) events. Other event distributions include TAG (5.98%), COMPLETE (5.31%), UNTAG (4.18%) and DELETE (2.17%). The frequencies of the remaining events (UNCOMPLETE, MOVE, COPY, FOLLOW, UNDELETE and UNFOLLOW) each lie below 1%.

When looking at the distribution of TM actions categorized according to CRUD as summarized in Table 5, it becomes clear that the consumption of content accounts for 51% of all events. The share of create (24%) and update (19%) events is similar, while the remaining 6% represent delete events.

As outlined, the share of Activities within communities is higher than the share of Activities outside of communities. Thus, in the following, we investigate the use of TM features in different types of communities (see section 5.1).

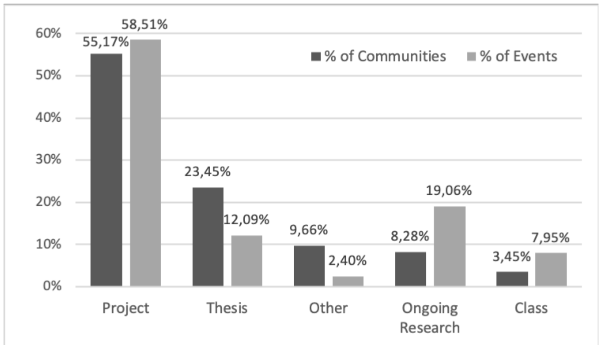

Figure 3: Distribution of community types

After filtering the data (section 5.2), 145 communities remained in which the Activities module has been used between 04/01/17-09/30/18 (3 semesters). The distribution of community types is as follows (see Figure 3): The majority of communities (55.17%) are project communities. In total, 58.51% of events in the Activities module are created within project communities. 23.45% of the communities are used to organize theses; however, only 12.09% of all events stem from thesis communities. This indicates that TM is not frequently used within these communities, as they serve the purpose of self-organization. This leads to a lower amount of collaboration inside the Activities module. While 8.28% of communities are used to conduct ongoing research, they produce 19.06% of all events. This shows that TM is used frequently in research communities compared to other community types. Only 3.45% of all communities are used for organizing classes. This can be explained with the nature of these communities and the lack of requirements for TM.

Table 10: Average number of active community members

Community Type	Total Members	Members in Activities
Class	54.6	13
Project	6.7	2.9
Thesis	3.9	1.6
Ongoing Research	13	4.3

Different community types can also be differentiated by their total number of active members and members that make use of TM functionalities (Table 10).

Class communities have the highest number of members. This is not surprising as these communities contain all students enrolled in a course, as well as the academic staff. There were only five communities of this type in the group of analyzed communities with two of them having more than 90 members and emphasizing the use of Activities. Thus, the average number of active community members using the Activities module is not representative for this community type

Project communities have an average of 6.7 active members, with about half of them being active users of the Activities module. Although in a typical project, all team members are usually involved in TM, the relatively low number can be explained. Most project communities are student project communities. The supervisors are active community members; however, they are not involved in the group's TM.

Thesis communities have an average of 3.9 members, which usually includes the student writing the thesis, the two supervisors as well as (optional) additional academic staff involved in the thesis. As the focus of these communities is self-organization, the average number of members being active in the Activities module is 1.6, i.e. the student and (optionally) a member of academic staff. Thus, thesis communities show a low collaborativity compared to other communities.

Ongoing research communities consist of 13 members on average. A third of them actively use Activities. This can be explained with the allocation of responsibilities inside a research group. Often, many academics are members of such communities, but they may not be actively involved in that area of research.

When looking at the events generated in different communities, it became apparent that many communities initially use the provided TM functionalities extensively; however, many of them stop using TM after a short time. Furthermore, TM functionalities seem to not have been used regularly, but rather on selected days (e.g. throughout a project, tasks are not completed when their execution has finished, but are rather "accumulated" and completed at a later point in time).

In order to verify this observation, the number of days on which events have been generated inside the Activities module have been extracted and evaluated. All days exclusively containing autocomplete events by the system have been omitted. The remaining communities have an average of 7.3 days of TM usage (median: 2), which supports the observation that the Activities module and its content are not updated on a regular basis. However, the mode of active days is 1 with a total amount of 50 communities (i.e., more than one third of all considered communities) that have only generated events on a single day. The initial adoption rate of TM functionalities seems rather high; however, many communities stop using the Activities module after one day and therefore do not reach the stage of task completion in the system.

6. Conclusion

In this research, we have investigated *coordination* on an academic collaboration platform by focusing on the use of TM features. We first established a common terminology and derived a TM catalog from academic literature. After mapping these functionalities to the academic collaboration platform UniConnect, we successfully demonstrated how SCA can be applied for measuring coordination in different types of virtual teams in ECS. Based on content data, we investigated the characteristics of tasks and based on transactional data, we demonstrated how different types of teams make use of TM functionalities.

The analysis included the characteristics and structure of content elements (e.g. attributes and information about hierarchies) as well as information about the usage and, therefore, collaboration itself.

Our main observation was that people do not seem to use TM functionalities consistently in UniConnect. This includes (1) people initially creating TM-related content but stopping to work on it after often not more than one day, and (2) people using TM to document a project over time, i.e. irregularly adding and updating content as opposed to making use of TM functionalities in real-time. In order to be able to understand the cause of this observation, it is necessary to further investigate user behavior. Unfortunately, the UniConnect databases do not yet provide enough data to conclude from comparing successful and unsuccessful TM behavior. An investigation of unsuccessful TM solely based on content and transactional data requires larger amounts of data, preferably from different ECS or TMS. As this is one of the first works that provide a holistic analysis by combining content and transactional data to analyze TM features in ECS, we also do not yet have sufficient reference results that allow classification and assessment of our results. Thus, future research will be directed towards expanding the scope of analysis to other systems. This will allow us to measure and derive implications for the success of coordination in virtual teams.

Furthermore, we were able to show that different types of communities could be further characterized regarding their use of TM functionalities. In this case, future research would include the analysis of specific communities in order to further analyze TM behavior both between and inside of different community types. Additionally, it would be interesting to direct further research towards different user groups (e.g., professors, students) and their TM-related behavior.

As mentioned previously, the TM Catalog is still at an early stage as it is primarily based on academic literature. It has only been applied once to evaluate the TM functionalities of UniConnect. Each application provides suggestions for minor revisions to refine the catalog. Thus, expanding the scope of analysis to other systems will also aid in improving the catalog.

This work solely focuses on conducting objective analyses of TM content and usage. Therefore, the immediate next step would be to extend this research with subjective analyses. In this case, user interviews could provide valuables insights into the use of TM functionalities. This could not only aid in gaining a better understanding of the reasons behind "unsuccessful" TM but also help to verify the findings resulting from SCA and thus increase the overall validity of this research.

7. References

[1] Bellotti, V., B. Dalal, N. Good, P. Flynn, D.G. Bobrow, and N. Ducheneaut, "What a To-Do: Studies of Task Management Towards the Design of a Personal Task List Manager", (2004), 735–742.

[2] Bellotti, V., N. Ducheneaut, M. Howard, and I. Smith, "Taking Email to Task: The Design and Evaluation of a Task Management Centered Email Tool", (2003), 345–352.

[3] Bellotti, V., N. Ducheneaut, M. Howard, I. Smith, and R.E. Grinter, "Quality Versus Quantity: E-Mail-Centric Task Management and Its Relation With Overload", *Human Computer Interaction 20*, 2005, pp. 89–138.

[4] vom Brocke, J., A. Simons, B. Niehaves, K. Reimer, R. Plattfaut, and A. Cleven, "Reconstructing the Giant: On the Importance of Rigour in Documenting the Literature Search Process", (2009), 1–12.

[5] vom Brocke, J., A. Simons, K. Riemer, B. Niehaves, R. Plattfaut, and A. Cleven, "Standing on the Shoulders of Giants: Challenges and Recommendations of Literature Search in Information Systems Research", *Communications of the Association for Information Systems 37*, 2015.

[6] Buehler, R., D. Griffin, and M. Ross, "Exploring the 'Planning Fallacy': Why People Underestimate Their Task Completion Times", *Journal of Personality and Social Psychology 67*(3), 1994, pp. 266–381.

[7] Campbell, C.S., and P.P. Maglio, "Supporting Notable Information in Office Work", (2003), 902–903.

[8] Chapman, P., J. Clinton, R. Kerber, et al., *CRISP-DM 1.0: Step-by-step data mining guide*, CRISP-DM Consortium, 2000.

[9] Chasanidou, D., B. Elvesæter, and A.-J. Berre, "Enabling team collaboration with task management tools", ACM Press (2016), 1–9.

[10] Conley, K., and J. Carpenter, "Towel: Towards an Intelligent To-Do List", (2006).

[11] Cozzi, A., S. Farrell, T. Lau, et al., "Activity management as a Web service", *IBM Systems Journal 45*(4), 2006, pp. 695–712.

[12] Craven, N., and D. Mahling, "Goals and processes: a task basis for projects and workflows", (1995), 237–248.

[13] Dillenbourg, P., "What do you mean by collaborative learning?", In *Collaborative-learning: Cognitive and Computational Approaches*. Elsevier, Oxford, United Kingdom, 1999.

[14] Dredze, M., T. Lau, and N. Kushmerick, "Automatically classifying emails into activities", ACM Press (2006), 70–77.

[15] Fan, S., X. Li, and J.L. Zhao, "Collaboration process patterns and efficiency of issue resolution in software development", *2012 International Conference on Collaboration Technologies and Systems (CTS)*, (2012), 559–565.

[16] Fan, S., X. Li, and J.L. Zhao, "Collaboration Process Pattern Approach to Improving Teamwork Performance: A Data Mining-Based Methodology", *INFORMS Journal on Computing 29*(3), 2017, pp. 438–456.

[17] Farrell, R., H. Shah, T. Erickson, and W.A. Kellogg, "A design evaluation of a user interface for tending long-term tasks", (2009), 4141–4146.

[18] Geyer, W., M.J. Muller, M.T. Moore, et al., "Activity Explorer: Activity-centric collaboration from research to product", *IBM Systems Journal 45*(4), 2006, pp. 713–738.

[19] Gil, Y., V. Ratnakar, T. Chklovski, P. Groth, and D. Vrandecic, "Capturing Common Knowledge about Tasks: Intelligent Assistance for To-Do Lists", *ACM Transactions on Interactive Intelligent Systems 2*(3), 2012, pp. 1–35.

[20] Glitsch, J.H., and P. Schubert, "IRESS: Identification of Requirements for Enterprise Social Software", (2017), 866–873.

[21] Gotta, M., N. Drakos, and M. Jeffrey, *The Future of Social Software in the Workplace*, Gartner, 2017.

[22] Haraty, M., D. Tam, S. Haddad, J. McGrenere, and C. Tang, "Individual differences in personal task management: a field study in an academic setting", *Graphics Interface*, (2012), 35–44.

[23] Harrison, B.L., A. Cozzi, and T.P. Moran, "Roles and relationships for unified activity management", ACM Press (2005), 236–245.

[24] Holz, H., O. Rostanin, A. Dengel, T. Suzuki, K. Maeda, and K. Kanasaki, "Task-based process know-how reuse and proactive information delivery in TaskNavigator", (2006).

[25] Jyothi, N.S., and A. Parkavi, "A Study on Task Management System", (2016), 1–6.

[26] Kamsin, A., A. Blandford, and A.L. Cox, "Personal task management: my tools fall apart when I'm very busy!", (2012), 1369–1374.

[27] Kato, J., D. Sakamoto, T. Igarashi, and M. Goto, "Sharedo: to-do list interface for human-agent task sharing", (2014), 345–351.

[28] Kreifelts, T., E. Hinrichs, and G. Woetzel, "Sharing To-Do Lists with a Distributed Task Manager", (1993), 31–46.

[29] Malone, T.W., and K. Crowston, "The interdisciplinary study of coordination", *ACM Computing Surveys 26*(1), 1994, pp. 87–119.

[30] Miles, M.B., A.M. Huberman, and J. Saldana, *Qualitative Data Analysis*, SAGE Publications Inc., 2014.

[31] Millen, D.R., M.J. Muller, W. Geyer, E. Wilcox, and B. Brownholtz, "Patterns of media use in an activity-centric collaborative environment", (2005), 879–888.

[32] Moran, T.P., "Activity: Analysis, design, and management", (2003).

[33] Moran, T.P., "Unified Activity Management: Explicitly Representing Activity in Work-Support Systems", (2005), 1–10.

[34] Mundbrod, N., and M. Reichert, "Process-Aware Task Management Support for Knowledge-Intensive Business Processes: Findings, Challenges, Requirements", (2014), 116–125.

[35] Poncin, W., A. Serebrenik, and M. v d Brand, "Process Mining Software Repositories", *2011 15th European Conference on Software Maintenance and Reengineering*, (2011), 5–14.

[36] Raposo, A.B., and H. Fuks, "Defining Task Interdependencies and Coordination Mechanisms for Collaborative Systems", *Cooperative Systems Design (vol 74 of Frontiers in Artificial Intelligence and Applications)*, 2002, pp. 88–103.

[37] Schmidt, K., and L. Bannon, "Constructing CSCW: The First Quarter Century", *Computer Supported Cooperative Work (CSCW) 22*(4), 2013, pp. 345–372.

[38] Schubert, P., and S.P. Williams, "The Case of UniConnect – The Shaping of an Academic Collaboration Platform", (2016), 327–338.

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

[40] Schwade, F., and P. Schubert, "Social Collaboration Analytics for Enterprise Social Software: A Literature Review", *Multikonferenz Wirtschaftsinformatik (MKWI)*, (2018).

[41] Siu, N., L. Iverson, and A. Tang, "Going with the flow: email awareness and task management", (2006), 441–450.

[42] Toxtli, C., A. Monroy-Hernández, and J. Cranshaw, "Understanding Chatbot-mediated Task Management", (2018), 1–6.

[43] Treude, C., and M.-A. Storey, "Bridging lightweight and heavyweight task organization: the role of tags in adopting new task categories", (2010), 231–234.

[44] Treude, C., and M.-A. Storey, "Work Item Tagging: Communicating Concerns in Collaborative Software Development", *IEEE Transactions on Software Engineering 38*(1), 2012, pp. 19–34.

[45] Wehner, B., C. Ritter, and S. Leist, "Enterprise social networks: A literature review and research agenda", *Computer Networks 114*, 2017, pp. 125–142.

[46] Williams, S.P., *Enterprise 2.0 and Collaborative Technologies*, University of Koblenz-Landau, Koblenz, Germany, 2011.

[47] Williams, S.P., and P. Schubert, "Connecting Industry: Building and Sustaining a Practice-based Research Community", *Proceedings of the 50th Hawaii International Conference on System Sciences*, (2017).

[48] Wohlin, C., "Guidelines for snowballing in systematic literature studies and a replication in software engineering", ACM Press (2014), 1–10.