Human-centric Computing and Information Sciences
a SpringerOpen Journal

RESEARCH Open Access

Task context-aware e-mail platform for collaborative tasks

Masashi Katsumata

Correspondence: katumata@nit.ac.jp
Department of Computer and Information Engineering, Nippon Institute of Technology, 4-1 Gakuendai, Miyashiro-machi, Minamisaitama-gun, Saitama, Japan

Abstract

E-mail communication has become an essential part of collaborative tasks in enterprises. However, when conventional e-mail applications are used for collaborative tasks, problems with task-related resources management arise. In this paper, we present a task context-aware e-mail platform that helps users to send e-mails quickly and efficiently. This platform also automatically extracts data from reply e-mail messages. It enables users to automatically classify task-related information and user support services using a task context model. The task context model is built based on ontology as a semantic representation of the associations between task and task-related e-mail processes. This paper describes the design and implementation of this system on the basis of the task context model. To verify the efficacy of the prototype system, we conducted experiments that demonstrate the systems effective task awareness and user support services.

Keywords: Task context; Ontology; Collaborative task; E-mail platform

Introduction

E-mail-based communication has become an essential part of collaborative tasks in enterprises. E-mail communication is required for organizational tasks in order to achieve effective task management and for reuse of e-mail messages and their related resources (such as schedules, attached files, and contact lists). Knowledge workers can efficiently search and use e-mail messages and the corresponding resources by organizing these messages according to individual tasks [1,2]. Thus, multi-tasking knowledge workers often set up automatic filtering into folders or manually move e-mail messages into folders. Recently, enhanced conventional task-management systems and task-centric mail clients have been used for this purpose. Further, several research studies that support the discovery of e-mail messages and the related resources by adding metadata to e-mails and the related resources have been conducted [2-6].

In this paper, we present a task context-aware e-mail platform that helps users to send e-mails quickly and efficiently. This platform also automatically extracts data from the e-mail messages returned in reply. In order to realize the concept of this platform, we introduce the task context model as an ontology-based semantic representation of the conceptual associations between a task and the task-related e-mail process. By using the task context model, the system can provide a context-aware service for mail form composition and mail data extraction. To verify the feasibility of the concept of this platform, we built a prototype system. We also describe the system's effective task awareness and user support services evidenced in the experimental results.

The remainder of this paper is organized as follows. First, we give an overview of related work. Then, we present the features of the task context model and discuss the design of the task context-aware e-mail platform. To verify the feasibility of the concept underlying this platform, we present a prototype implementation and discuss its effective task awareness and user support services evident in experimental results obtained. Finally, we conclude our paper and outline future research directions.

Related work

In using e-mail for organizational work, a vast amount of task-related information has to be handled. Therefore, much research has been done in the area of task-related management support in the past few years.

TaskMaster [3] enhances e-mail clients and enables them to function as task-management systems by managing resources such as e-mail messages and file attachments for each task. In addition, a useful user interface with both browsing and operating resources is also provided. This system makes it easy to search through the resources of a task. TV-ACTA [4] provides prestructured containers that are created inside the e-mail folder hierarchy to support personal information management. Specialized subfolders called "Components" within each ACTA Activity automatically organize and present information appropriate for aspects of the activity at hand. ACTA is designed to create a more efficient personal information management environment with the ultimate goal of providing context metadata for machine learning and automation techniques.

KASIMIR [7] and OntoPIM [8] are ontology-based personal task-management systems. These systems provide semi-automated functions for retrieving and registering task-related information within e-mail messages according to an ontology-based model. Activity Explorer [4] supports knowledge workers with context switching and resource rediscovery by organizing and integrating resources, tools, and people around the computational concept of a work activity. However, the support functions of the systems presented above do not apply to reusing of the managed data for tasks.

Eklund and Cole [9] and Brendel and Krawczyk [10] used ontology to model e-mail-related attribution information (such as group, project, and member), and proposed a system that provides a user interface with visualized and grouped formal concept analysis that can be used to search for various types of task-related information. Topika [11] enhances an existing e-mail client to provide suggestions about relevant shared spaces such as Wikis. The system facilitates the transition management of a user's collaborative activities to appropriate collaboration tools.

These systems are all primarily concerned with improving the management of and the search for task-related information. In contrast, our primary goal is to provide support for reuse of managed data according to a user's role.

Design of the task context-aware e-mail platform

Task context model

We created an ontology-based semantic representation model that represents the conceptual associations between a task and e-mail processes. We call this the task context model. The task context-aware e-mail platform performs services for users based on this task context model. The model relates the conceptual associations between a task and an e-mail process to physical context entities (such as e-mail messages, attached files, group members, and mail form items). In the task context model, task-related e-mail messages and resources, called task context data, are handled as a task unit. The files created and schedule data for the given task are also handled as task context data. In addition, this platform provides a service that automatically retrieves the task-related resources contained in e-mail messages. The operation of this service is considered later in this paper. These task context data are utilized in the user support service to accomplish tasks.

The semantic representation of the task context model is based on the Resource Description Framework (RDF) [12]. RDF is a collection of triples, each of which consists of a resource, a property, and a literal. A set of such triples is called an RDF graph. Figure 1 shows a sample task context model represented by an RDF graph.

The task context concept represents the role of managing task-related resources. In addition, on the assumption that it is used within an organization, the task context concept represents the relationship among task-related resources that are frequently used in collaborative work. The task context is represented as a property of a "Task" resource. We define the conceptual associations of the task context's attribution as follows: Task is "Subject", task context is "Predicate", and the value of the task context denotes "Literal". In the task context model, task contexts are classified as entities such as files, schedules, participants, memos, and mail form items. Further, we define the conceptual mail procedure in collaborative task. We call this "Action". The Action concept represents mail procedures that frequently occur in collaborative task e-mail communications. We consider the following three Actions for collaboration tasks:

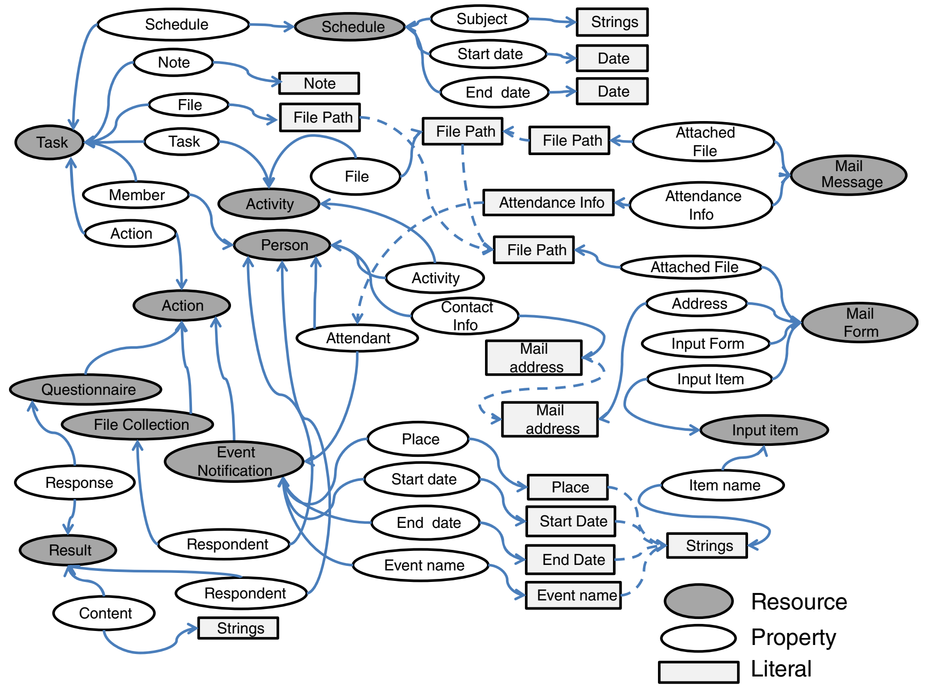

Figure 1 **Sample task context model represented by an RDF graph.**

1. Event Notification. A mail procedure concept that has the objective of attendance confirmation for an event being held in an organization.
2. Questionnaire Request. A mail procedure concept that has the objective of soliciting questionnaire requests and responses.
3. File Collection. A mail procedure concept that notifies that a file is to be collected and obtains an attached file in response.

Task context management

The task context model manages the task context's property and its value for each task. This property represents a task-related file, member, contact information, and schedule data, as shown in Figure 1. The value of the task context is automatically retrieved from an e-mail message on the basis of the task context model when the e-mail arrives on the mail server implemented in our study (see Figure 2).

Figure 3 shows how task context is assigned to property in the task context model. When the task owner registers the task for group work in the client, RDF/XML formatted data based on the task context model are automatically generated and managed by the task context server. The upper part of Figure 3 shows the RDF model of the task context model, and the lower part shows the task context data that are generated according to the task context model. The figure also shows how this model can concurrently handle multiple tasks such as "meeting" and "lab party".

Service for e-mail process

The aims of the task context-aware e-mail platform are to support 1) the composition of e-mail forms and 2) the extraction of the data contained in reply e-mails. The support for creating e-mail forms is provided when a task owner creates an e-mail form

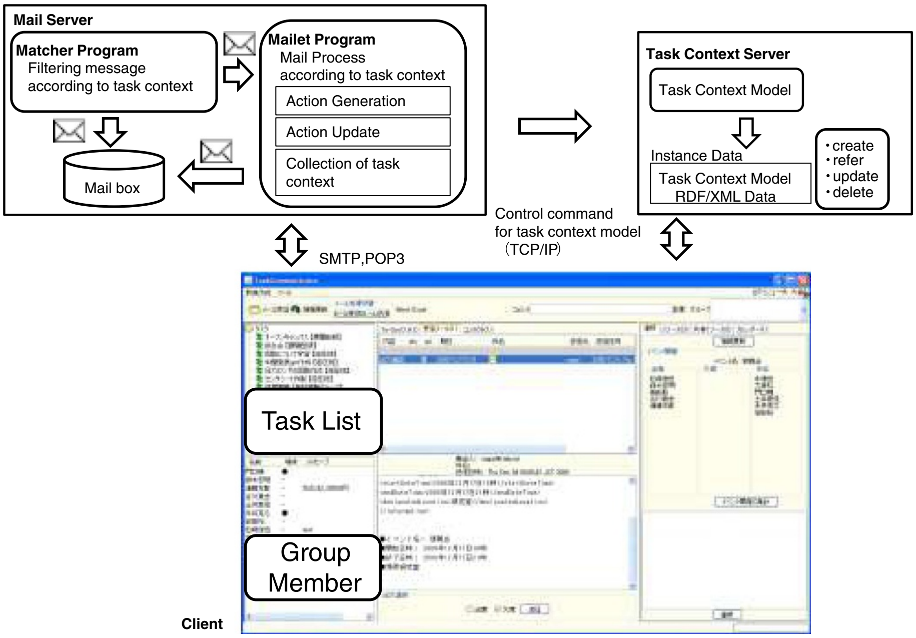

Figure 2 Task context-aware e-mail platform.

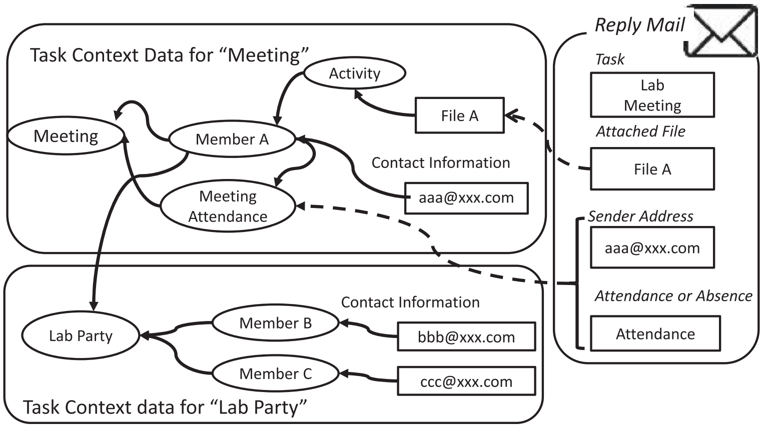

Figure 3 **Task context model-based information management.**

for a task request and a task member creates an e-mail form in reply to a task request. On the other hand, the support for the extraction of data contained in a reply e-mail is provided when a task owner receives a reply mail from a task member. In this paper, the user support service is intended for three actions: Event Notification, Questionnaire Request, and File Collection. The user support service in the e-mail process provides a service that supports both the creation of e-mail forms and organization and retrieval of data from reply e-mails for these mail actions.

Implementation

System overview

We implemented a prototype system that executes a service for task members on the task context-aware e-mail platform. The prototype system comprised the following three systems: task context server, mail server, and mail client (see Figure 2). The task context server manages the task context (e.g., file path, schedule, and contact information) and its value, which is generated according to the RDF/XML data format. In the task context server, task context is managed under each task. We implemented a task context server that can connect to the mail server and the mail client. The task context server can accept a request command (create, refer, update, and delete) from the mail server and client via TCP/IP. On accepting a request command, the task context server can update the task context model using the Jena application programming interface [13]. The mail server is built on Apache James [14]—a mail application platform that enables users to write custom application programming code for e-mail processing. Apache James provides e-mail filtering through a function called Matcher and provides e-mail processing through another function called Mailet. We introduced extended e-mail headers to realize the service (see Table 1). After Matcher refers to the extended e-mail headers, Mailet can be executed according to the purpose of these extended e-mail headers.

The client can connect to both the mail server and the task context server. In addition to general e-mail operations, the client provides a user interface that manages the task context data. The e-mail message submitted by the client is automatically added to the extended e-mail header. In our prototype system, the client displays a structured mail form by referring to the extended e-mail header. The client can also

Table 1 Types of extended mail headers

Header name	Role
X-Task-Name	Indicates the name of the task
X-Task-Owner	Indicates the task owner
X-Action-Model-Type	Indicates the request generating the Action and the type of Action
X-Action-Update-Type	Indicates the request updating the state of Action and the type of Action
X-Action-Retrieve-Type	Indicates the request retrieving the task context data and the type of Action

receive other e-mail messages in accordance with RFC2822 [15]. The client's user interface comprises six main areas—task member, file, calendar, task, message, and form. The prototype system was implemented in Java, using Apache James to run the mail application platform and to handle XML messages.

E-mail form composition service

When a task owner selects the type of Action on the client, an e-mail form for the selected Action is displayed. In the e-mail form for the Action, task-related data are provided as a list of suggestions of possible inputs. Consequently, the task owner spends less time typing and querying for information related to the task. When the task client receives an e-mail from the task owner, the reply form is displayed by referring to the extended e-mail header X-Action-Model-Type.

The displayed input fields on the reply form are the elements of the Action type corresponding to "reply mail form composition," as shown in Table 2. A task member can type the value according to the displayed input field on the reply form.

Data extraction service

When the mail server receives an e-mail according to the type of Action, the contents of the reply e-mail are automatically retrieved as the task context. In the task context server, the retrieved task context data are managed as RDF/XML formatted data that are based on the conceptual model for Action. When the task context server receives a reply e-mail, the value of the task context in Action is updated, and the state of Action is displayed on the client's state panel. Thus, the task owner can confirm the state of the task intuitively without checking each reply e-mail. Automated processes such as the generation or updating of Action are performed via Mailet according to the value of the extended mail header.

Table 2 Services for action type

Function	Event notification	Questionnaire request	File collection
Request mail form composition	Schedule form (Start date, End date, Place, Event name)	Questionnaire form (Support form composition)	File name and stored folder name
Request mail form composition	Attendance form (Yes, No)	Questionnaire form (reply form)	Selection of attached file
Automated extraction	Attendance data	Questionnaire response data	Attached file

The process of generation of an Action instance is depicted in Figure 4. When the mail server receives an e-mail from the task owner, Matcher program determines Mailet program according to the value of extended e-mail header. Mailet program generates a task context model according to the given Action on the task context server. When the mail server receives the reply e-mail message from the task member, the Matcher program refers to the extended e-mail header X-Action-Update-Type, and Mailet program updates the state of Action. Mailet program updates the value of state by referring to the text element in XML format contained in the body part of the e-mail message. The above procedure is also followed for reply e-mails from other task members and updates to the value of the attendance for Action. The attendance data and the questionnaire data can be written to a Comma Separated Value (CSV) file on the assumption that task context data might be used by a spreadsheet application (e.g., Microsoft Excel). In the case of File Collection, files are automatically renamed according to a predefined file name from the attached file in the task member's reply e-mail.

Evaluation

Experimentation overview

We conducted an experiment to verify the efficacy of our platform in which we obtained qualitative data compiled from 13 university students (male, 21–22 years old). The prototype system was set up in our laboratory. One of the 13 students was elected as the task owner, and the others as task members. We conducted the experiment according to the following procedure. First, the task owner notified the task members about a group meeting for a research report. On receiving the e-mail about a group meeting, task members replied to the e-mail stating attendance of meeting to the task owner. After the group meeting, the task owner requested completed questionnaires from the task members and the meeting report for a research presentation. The task owner then checked and confirmed that reply e-mails were received from the task members. Further, he gathered or added up the data in the messages within the reply e-mail.

The experiment was divided into two sessions: using a conventional e-mail platform and using our prototype system. The two sessions were administered sequentially. During the conventional e-mail platform session, the procedures for the experiment were held over the first four weeks. In the next four weeks, the procedures for the experiment using the prototype system were held. In this experiment, we made the following assumptions:

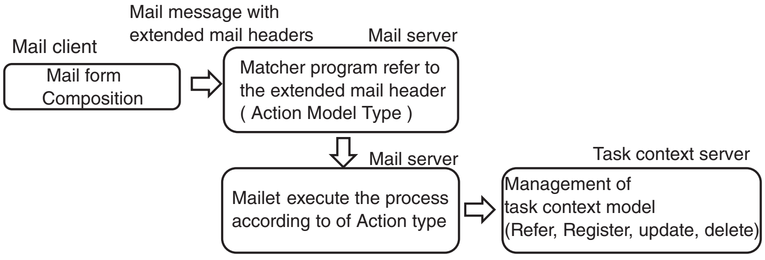

Figure 4 Process for action instance on the prototype system.

1. Conventional e-mail system scenario
   (a) Task owner's usage condition and work
      (i) The e-mail addresses of the task owner and task members are registered in the e-mail client.
      (ii) The date of the group meeting is provided to the task members.
      (iii) Indication of attendance confirmation, questionnaire request, and deadline of the research report are handled by e-mail, and the task owner checks the reply e-mails.
   (b) Task member's usage condition and work
      (i) The reply e-mail is created in response to the task owner's request e-mail.
2. Proposed e-mail system scenario
   (a) Task owner's usage condition and work
      (i) Each group member's e-mail address is registered in the proposed system.
      (ii) The scheduled date for the group meeting is registered in the scheduler of the client system.
      (iii) Indication of attendance confirmation, questionnaire request, and deadline of the report are handled by e-mail. Collecting the content for the reply e-mail is performed by the user support service in the proposed system.
   (b) Task member's usage condition and work
      (i) A reply e-mail is created in response to the task owner's request e-mail.

Experimental results

This section presents the experimental results obtained in order to verify the efficacy of the prototype system and its provision of the service that indicates the task context according to the user's role. By means of an experiment, we realized our concept using a prototype system. Then, we investigated the results of the questionnaire regarding the prototype system's superiority compared with the conventional e-mail client (Mozilla Thunderbird).

Table 3 shows the results of the questionnaire in terms of task awareness for the following user support services: Event Notification, Questionnaire Request, File Collection, and Task context management.

We describe the concern of task awareness for user support services below.

Awareness of task

The task owner was rated more than 4.0 on average for each category (Event notification, Questionnaire Request, File Collection). We believe that this assessment of the user support service for the task owner was given a high evaluation because of the automated function for the task owner. In checking attendance for meeting, the

Table 3 Results of the questionnaire about usage of the prototype system

Action type	Task owner		Task member	
	Awareness of task	Awareness of task-related resources	Awareness of task	Awareness of task-related resources
Event notification	4.75	4.75	4.5	4.3
Questionnaire request	4	5	4.7	4
File collection	4.5	4.75	4.8	4.27

following opinion was obtained: “It is good because of the possibility to check the state of the task on the client”.

The task member rates were more than 4.5 on average. Hence, it appears that task members could easily notice newly arrived tasks. Because task information is managed in the task context server, it pushes new task information to the user’s client automatically. We think that the assessment of the work necessary for the reply e-mail attained a high valuation.

Awareness of task-related resource

The task owner was rated more than 4.75 on average for each category. Thus, it appears that our prototype system reduced the number of search operations for task-related information compared with the conventional e-mail system. Our prototype system provides an automated function for the task owner. The automated function (implemented in the Mailet program) extracts the context data from the reply e-mail message. The task owner can thus check the task-related resources in each panel shown in Figure 2. These reasons strongly substantiate the high rating obtained from the task owner. We found that the user interface view for the task owner was effective for organizing task-related information. Moreover, we also found that the burden of operation for organizing task-related information decreased when performing user support service.

The task member rates were more than 4.2 on average for each category. To provide the context data for a given task, we implemented task context view panel on the client shown in Figure 2. Therefore, task members could easily confirm the task schedule on the calendar panel. For each category, the results confirmed a high valuation in terms of awareness of service functionality for a given task.

Conclusions

In this paper, we described the design and implementation of a task context-aware e-mail platform for collaborative tasks. In order to provide a task context-aware service for task members, we introduced a task context model that represents conceptual associations between a task and the related mail process. Using the prototype system, we confirmed that the task context-aware platform executed the required services on the basis of the task context model. An operational experiment conducted enabled us to obtain insightful comments that can help to improve the prototype system for practical use from the point of view of an actual user environment. In future work, we will further develop the prototype to support more tasks.

The prototype system was deployed and evaluated in the relatively small confines of our laboratory, resulting in only a limited number of tasks and e-mail messages being handled among group members. Therefore, to validate our concept, we will expand the prototype system uses to larger organizations.

Competing interests

The author declares that he has no competing interests.

Received: 30 January 2014 Accepted: 1 September 2014
Published online: 20 September 2014

References

1. Krämer J-P (2010) PIM-Mail: Consolidating Task and Email Management. Extended Abstracts on Human Factors in Computing Systems. Proceedings of the 28th of the international conference extended abstracts on Human factors in computing systems, ACM, New York, pp 4411–4416
2. Ducheneaut N, Bellotti V (2001) Email as a Habitat. In: An Exploration of Embedded Personal Information Management, ACM Interactions, 8., pp 30–38
3. Bellotti V, Ducheneaut N, Howard M, Smith I (2003) Taking Email to Task: The Design and Evaluation of a Task Management Centered Email Tool. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. ACM, New York, pp 345–352
4. Bellotti V, Thornton JD, Chin A, Schiano D, Good N (2007) TV-ACTA: Embedding an Activity-Centered Interface for Task Management in Email. In: Proceedings of the Conference on Email and Anti-Spam
5. Geyer W, Muller MJ, Moore MT, Wilcox E, Cheng LT, Brownholtz B, Hill C, Millen DR (2006) Activity explorer: activity-centric collaboration from research to product. IBM Syst J 45:713–738
6. Cozzi A, Farrell S, Lau T, Smith BA, Drews C, Lin J, Stachel B, Moran TP (2006) Activity management as a web service. IBM Syst J 45:695–712
7. Grebner O, Ong E, Riss UV (2008) KASIMIR—Work Process Embedded Task Management Leveraging the Semantic Desktop. In: Proceedings of Multikonferenz Wirtshaftsinformatik, Workshop Semantic Web Technology in Business Information Systems., pp 715–726
8. Lepouras G, Dix A, Katifori T, Catarci T, Habegger B, Poggi A, Ioannidis Y (2006) OntoPIM: From Personal Information Management to Task Information Management. In: Proceedings of SIGIR Workshop on Personal Information Management., pp 78–81
9. Eklund P, Cole R (2002) Structured Ontology and Information Retrieval for Email Search and Discovery. In: Hacid M-S, Raś ZW, Zighed DA, Kodratoff Y (eds) ISMS2002, 2366. Proceedings of the 13th International Symposium on Methodologies for Intelligent Systems (ISMIS 2002), Lyon, France, pp 75–84
10. Brendel R, Krawczyk H (2008) E-mail User Role Identification Using OWL-Based Ontology Approach. In: Proceedings of the 1st International Conference on Information Technology., pp 18–21
11. Mahmud J, Matthews T, Whittaker S, Moran TP, Lau T (2011) Topika: Integrating Collaborative Sharing with Email. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. ACM, New York, pp 3161–3164
12. Candan KS, Liu H, Suvarna R (2001) Resource description framework: metadata and its applications. ACM SIGKDD Explorations Newsletter. ACM, New York, 3(1):6–19
13. Jena API https://jena.apache.org/
14. Apache James Project http://james.apache.org
15. Resnick P (2001) Internet Message Format. RFC Editor

doi:10.1186/s13673-014-0017-7
Cite this article as: Katsumata: **Task context-aware e-mail platform for collaborative tasks.** *Human-centric Computing and Information Sciences* 2014 **4**:17.